LUISS Guido Carli
LIBERA UNIVERSITÀ INTERNAZIONALE DEGLI STUDI SOCIALI

Dipartimento di Impresa e Management
Tesi di laurea triennale

Cattedra
STATISTICA APPLICATA ED ECONOMETRIA

Sentiment analysis come strumento di previsione dei mercati finanziari

RELATORE
Prof. Giorgia Rivieccio

CANDIDATO
Vincenzo Camerlengo
Matr. 254191

Anno accademico 2022/2023

INDICE

INTRODUZIONE

Il presente lavoro si prefigge l'obiettivo di verificare l'esistenza di una correlazione tra i rendimenti dei titoli azionari ed il *sentiment* ad essi associati. In altri termini, la domanda alla quale si intende rispondere è: "si possono prevedere i rendimenti futuri dei titoli azionari utilizzando tecniche di intelligenza artificiale quali la *sentiment analysis*?"
Per rispondere a questa domanda prenderemo in esame tre titoli azionari quotati al NASDAQ, quindi in dollari statunitensi: AMZN (Amazon), TSLA (Tesla), AAPL (Apple).
La scelta è ricaduta su queste tre aziende data la loro grande influenza e l'ingente quantità di news che è possibile reperire al fine di costruire il modello più preciso possibile.

Nel primo capitolo approfondiremo la cosiddetta "Teoria dei Mercati Efficienti", sviluppata dagli economisti Eugene Fama e Paul Samuelson, secondo la quale è impossibile prevedere i rendimenti dei titoli. La loro argomentazione principale si basa sull'efficienza informativa, e cioè sul fatto che dato che tutti gli operatori del mercato sono razionali, tutte le notizie saranno immediatamente incorporate nei prezzi dei titoli non lasciando opportunità di guadagno sistematico. Passeremo poi ad analizzare tutte le critiche all'*EMH Theory* ed il conseguente avvento della "Teoria della Finanza Comportamentale", focalizzandoci sulla critica all'ipotesi di razionalità e sulla critica all'ipotesi di arbitraggio, per poi esaminare le cosiddette anomalie di mercato. Infine, introdurremo il *sentiment* ed il ruolo che ricopre all'interno di un'analisi di questo tipo.

Nel secondo capitolo esamineremo più tecnicamente il funzionamento della *sentiment analysis* e come fanno i *software* ad estrapolare il tono e l'emotività che si evince da un testo. Verrà spiegato quindi cos'è il *text mining* e cosa si intende per *Natural Language Processing* facendo anche un piccolo *excursus* storico partendo dagli anni '50 – '60 con la creazione della prima chatbot "ELIZA" fino ad arrivare ai giorni nostri. Nell'ultimo paragrafo di questo capitolo verrà anche introdotto l'algoritmo di Bayes con ottimizzato per la *sentiment analysis.*

Nel terzo capitolo analizzeremo *in primis* come il mondo dei *social media* sta influenzando le nostre vite ed in particolare come ha rivoluzionato il modo di diffondere le notizie. Entriamo poi, finalmente, nel vivo della ricerca spiegando il funzionamento e l'utilità di Refinitiv Workspace e Google Trends. Grazie al primo ho potuto estrapolare i dati del sentiment ed i rendimenti quotidiani dei titoli, mentre grazie al secondo ho ottenuto i volumi di ricerca su Google. Anche questi ultimi infatti possono rappresentare un ottimo predittore dei rendimenti dei titoli per la logica secondo la quale se molti utenti ricercano un titolo su Google, è molto probabile che siano interessati a scambiarlo sul mercato.

Infine, grazie all'utilizzo di Excel, è stato elaborato il dataset contenente le serie storiche dei rendimenti, del *sentiment*, e di Google Trends.

Nel quarto ed ultimo capitolo vengono illustrate le metodologie utilizzate e l'analisi empirica svolta con l'ausilio di R, volta alla creazione di modelli autoregressivi.
Infine, analizzando i risultati ottenuti, verificheremo se il *sentiment* può essere considerato un buon predittore dei rendimenti dei titoli azionari.

CAPITOLO PRIMO

È POSSIBILE PREVEDERE IL MERCATO AZIONARIO?

1.1 TEORIA DEI MERCATI EFFICIENTI

L'ipotesi dei mercati efficienti (Efficient Market Hypothesis, in sigla EMH) rappresenta uno dei fondamenti dell'economia finanziaria. Tale teoria è stata sviluppata a partire agli anni '60 dagli economisti Eugene Fama e Paul Samuelson presso l'Università di Chicago, i quali hanno elaborato le ipotesi che si trovano alla base dei mercati cosiddetti efficienti.

I fondamenti teorici dell'EMH si basano su tre argomentazioni (Shleifer, 2000)[1]:

- In primo luogo, si presume che gli investitori siano razionali e che quindi valutino ogni titolo per il proprio valore effettivo, cioè il valore attuale netto dei suoi flussi di cassa futuri, scontati in base alle loro caratteristiche di rischio.
  Pertanto, indicando con:

  P_t prezzo dell'azione al tempo t
  D_{t+1} dividendi attesi al tempo t+1
  D_{t+2} dividendi attesi al tempo t+2
  r_t tasso di interesse a un anno, al tempo t
  r_{t+1} tasso di interesse a un anno, atteso al tempo t+1
  ε premio al rischio

  Il prezzo del titolo al tempo t può essere espresso dalla relazione seguente:

  $$P_t = \frac{D_{t+1}}{(1+r_t+\varepsilon)} + \frac{D_{t+2}}{(1+r_t+\varepsilon)(1+r_{t+1}+\varepsilon)} + \cdots + \frac{D_{t+n}}{(1+r_t+\varepsilon)\ldots(1+r_{t+n-1}+\varepsilon)} + \cdots$$

  Quando gli investitori apprendono qualcosa riguardo i valori fondamentali di un titolo, risponderanno immediatamente aumentando i prezzi quando le notizie sono positive e, viceversa, diminuendoli se le notizie sono negative; di conseguenza i prezzi dei titoli incorporano quasi immediatamente le nuove informazioni disponibili. Come affermato da Fama, la razionalità degli investitori implica anche l'impossibilità da parte di quest'ultimi di ottenere rendimenti superiori per un dato livello di rischio.

[1] Shleifer, A. (2000). *Inefficient markets: An introduction to behavioral finance*. Oup Oxford.

- In secondo luogo, si presume che nella misura in cui alcuni investitori non siano razionali, le loro operazioni si annullano a vicenda senza influenzare i prezzi di equilibrio. Tale ipotesi si basa sul presupposto che le strategie di trading degli investitori irrazionali non siano correlate tra loro e che quindi siano casuali.

- In terzo luogo, quando gli investitori sono irrazionali in modo simile, si presume che interverranno sul mercato gli arbitraggisti che, essendo razionali, elimineranno l'influenza dei primi sui prezzi. L'arbitraggio è definito come "l'acquisto e la vendita simultanea dello stesso titolo, o di un titolo essenzialmente simile, in due mercati diversi a prezzi vantaggiosamente diversi" (Sharpe e Alexander, 1990)[2]. Per semplificare il concetto facciamo un esempio pratico. Supponiamo che un titolo subisca un sovrapprezzo rispetto al suo valore reale a causa di acquisti correlati da parte di investitori irrazionali. Notando questo eccesso di prezzo, gli investitori razionali cioè gli arbitraggisti, venderanno questi titoli per acquistarne di simili con lo stesso livello di rischio ottenendo così un profitto sicuro. La compravendita di titoli da parte degli arbitraggisti contribuirà quindi a far diminuire il prezzo dei titoli sopravvalutati o aumentare il prezzo di quelli sottovalutati, facendoli ritornare al prezzo di equilibrio.

L'intuizione sulla quale si basa l'intera teoria è perfettamente descritta da Fama il quale afferma che un mercato finanziario è detto efficiente solo se in ogni istante il prezzo delle attività scambiate riflette pienamente le informazioni rilevanti disponibili, per cui non sono possibili ulteriori operazioni di arbitraggio: la concorrenza garantisce che i rendimenti delle attività siano ai loro livelli di equilibrio (Fama, 1970)[3].

Una diretta implicazione dell'affermazione di Fama è che, dato che i prezzi sono influenzati unicamente dalle informazioni disponibili, le quali sono ovviamente imprevedibili, anche i prezzi seguiranno un andamento casuale. Pertanto, un investitore qualsiasi non può sperare di battere costantemente il mercato e le risorse che esso dedica all'analisi, alla selezione e alla negoziazione dei titoli sono sprecate.

Il contesto di mercato efficiente si esplicita attraverso tre forme di efficienza:

- **Efficienza allocativa**: tale efficienza si realizza se tutti gli operatori agissero in maniera razionale, ricercando le opportunità di investimento o finanziamento che consentono di massimizzare la loro utilità. Questo concetto si sovrappone a quello di ottimo paretiano, che

[2] Sharpe W. F. Alexander G. J. & Bailey J. V. (1990). *Investments William f. Sharpe Gordon j. Alexander fourth edition: instructor's manual*. Prentice Hall.
[3] Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The journal of Finance*.

si raggiunge quando non è possibile una riorganizzazione della produzione che migliori le condizioni di tutti. In questa situazione, quindi, nessuna persona può migliorare la propria condizione senza che qualcun altro peggiori la sua.

- **Efficienza tecnico-operativa**: questa efficienza si raggiunge agevolando gli scambi e diminuendo i costi di transazione per gli operatori di mercato. Un mercato è quindi efficiente da questo punto di vista se permette ai trader di procedere con gli scambi nel momento in cui ne hanno bisogno, al prezzo che preferiscono e per le quantità richieste.

- **Efficienza informativa**: tale efficienza si realizza quando i prezzi riflettono tutta l'informazione disponibile in un determinato momento. In un mercato efficiente in senso informativo gli investitori cercano di ottenere profitti da tutte le informazioni in loro possesso. In questo modo, le informazioni, che arrivano sul mercato in maniera casuale, sono immediatamente incorporate nei prezzi dei titoli e le opportunità di profitto sono eliminate. Quando è presente tale efficienza sono impossibili fenomeni come le bolle speculative, se non informa estremamente piccola, in quanto il valore di mercato dei titoli non si discosterebbe mai troppo da quello reale.

Posti i fondamenti teorici della teoria dei mercati efficienti, passiamo ora a descrivere gli aspetti empirici principali emersi tra gli anni '60 e '70 (Shleifer, 2000)[4]. Questi possono essere riassunti in due parti:

- In primo luogo, quando giungono al mercato notizie riguardanti il valore di un titolo, il prezzo dovrebbe reagire ed incorporare queste notizie "presto" e "correttamente" dove "presto" sta a significare che chi legge la notizia su un giornale, quindi ne viene a conoscenza in ritardo rispetto al mercato, non deve essere in grado di approfittare dell'informazione, e "correttamente" sta a significare che l'aggiustamento dei prezzi dovrebbe essere mediamente accurato evitando quindi che i prezzi sotto-reagiscano o sovra-reagiscano a particolari annunci.

- In secondo luogo, poiché il prezzo di un titolo deve rispecchiare il suo valore, questo non dovrebbe variare in assenza di notizie rilevanti sul titolo.

[4] Shleifer, A. (2000). *Inefficient markets: An introduction to behavioral finance*. Oup Oxford.

La reazione rapida e accurata dei prezzi dei titoli, così come la non reazione in assenza di informazioni, sono le due grandi previsioni della EMH theory.
La principale conseguenza della reazione veloce e corretta del prezzo alle nuove notizie, è che un'informazione "vecchia" non serve a far soldi. Il "far soldi" significa avere un profitto maggiore di quanto prevede il rischio (Fama, 1970)[5].
Da qui si evince il ruolo fondamentale che le informazioni svolgono all'interno di questa teoria.
In un suo articolo, Fama[6], descrisse tre tipi di informazione e distinse l'ipotesi dei mercati efficienti in altrettante tipologie:

- **Forma debole**: il set informativo a disposizione degli investitori è costituito solamente dai prezzi e dai rendimenti passati dei titoli. La forma debole inoltre prevede che, dato un certo livello di rischio, sia impossibile ottenere extra-profitti basandosi solo sulla conoscenza dei prezzi e dei rendimenti passati. Sotto l'ipotesi di neutralità del rischio, questa versione dell'EMH si riduce all'ipotesi del *random walk*, ovvero all'affermazione che i rendimenti azionari sono del tutto imprevedibili sulla base dei rendimenti passati (Fama, 1965)[7]. Tuttavia, non è possibile escludere che, in determinati periodi, alcuni investitori possano ottenere rendimenti positivi su determinati titoli. Tali rendimenti positivi si compensano però con quelli negativi con il risultato che, in media, non è possibile ottenere degli extra-rendimenti. L'efficienza in forma debole, quindi, non implica che non esista alcun analista in grado di ottenere extra-rendimenti, quanto piuttosto che nella media gli investitori non sono in grado di ottenere sistematicamente profitti.

- **Forma semi-forte**: il set informativo a disposizione degli investitori è costituito, oltre che dalle informazioni disponibili nella forma debole, anche da tutte le informazioni pubbliche (anche di tipo previsionale), riguardo i titoli. In modo quasi analogo alla forma descritta precedentemente, anche la forma semi-forte prevede che, dato un certo livello di rischio, sia impossibile ottenere extra-rendimenti basandosi solo sulla conoscenza delle informazioni storiche e delle informazioni pubbliche. Tale impossibilità è giustificata dal fatto che ogni qual volta un'informazione diventa di dominio pubblico, quest'ultima verrà immediatamente incorporata nel prezzo del titolo. Tuttavia, anche nella forma semi-forte è possibile che un investitore riesca ad ottenere extra-profitti se in conoscenza di informazioni interne non ancora divulgate al pubblico.

[5] Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The journal of Finance*.
[6] Fama, E. F. (1965). The behavior of stock-market prices. *The journal of Business*.
[7] Fama, E. F. (1965). The behavior of stock-market prices. *The journal of Business*.

- **Forma forte**: il set informativo a disposizione degli investitori è pressoché completo, in quanto costituito oltre che dalle informazioni storiche e pubbliche, anche da quelle private e interne alle aziende. La forma forte dell'EMH prevede l'impossibilità assoluta, escludendo ogni tipo di eccezione, di ottenere extra-rendimenti in quanto le informazioni interne, trapelando molto rapidamente e venendo subito incorporate nei prezzi, non possono essere sfruttate dagli investitori in maniera vantaggiosa.

La maggior parte degli studi sull'EMH theory si sono concentrati sulla forma debole e semi-forte. Riguardo la forma debole, Fama afferma che i prezzi delle azioni seguono dei percorsi casuali e che quindi non è possibile prevedere l'andamento di un titolo conoscendo l'andamento dei giorni passati. In altre parole, in un dato giorno, il prezzo di un'azione ha la stessa probabilità di scendere o salire a prescindere dal suo andamento il giorno precedente.

Allo stesso modo, si è testata la forma semi-forte dell'efficienza vedendo cosa accade nel breve periodo dopo che sono trapelate alcune notizie riguardo un'impresa come l'annuncio di distribuzione dei dividendi, la cessione dell'azienda, l'emissione di azioni ecc. In un loro studio, Keown e Pinkerton[8], hanno dimostrato che il prezzo delle azioni comincia a crescere prima dell'annuncio dell'offerta, come se la notizia di una possibile offerta fosse già incorporata nel prezzo, e poi effettua un salto il giorno dell'annuncio pubblico, senza seguire un'ascesa continua.

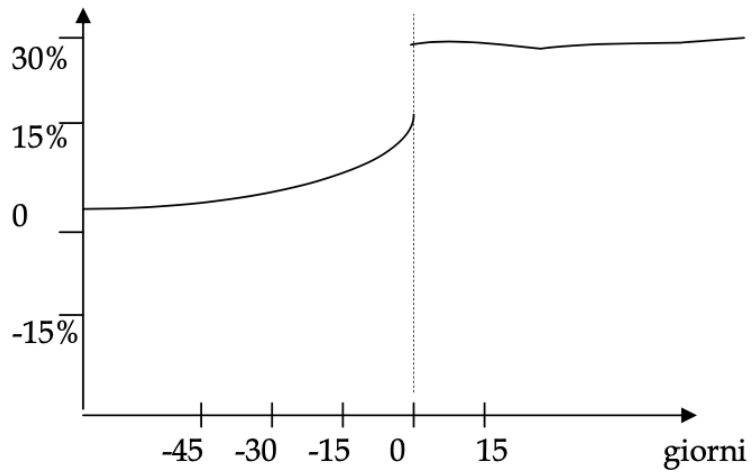


Fig.1 - Keown, A. J., & Pinkerton, J. M. (1981).

[8] Keown, A. J., & Pinkerton, J. M. (1981). Merger announcements and insider trading activity: An empirical investigation. *The journal of finance*.

1.2 CRITICHE ALL'EMH THEORY

Nonostante le innumerevoli tesi a supporto della teoria dei marcati efficienti, ci sono stati nel corso del tempo molti studiosi che si sono discostati da questo pensiero. Infatti, al giorno d'oggi si reputa la EMH theory vecchia ed ormai superata dalla "Teoria della Finanza Comportamentale".

La Finanza Comportamentale (*Behavioral Finance*) è un approccio alla finanza sviluppatosi intorno agli anni '70 del secolo scorso che utilizza gli studi della psicologia[9] e della sociologia[10] del comportamento degli individui, per comprendere le anomalie che si verificano nel mercato di capitali. Questa nuova teoria si basa sul presupposto che i traders, in quanto esseri umani, non operano sempre con razionalità e che quindi i prezzi dei titoli possono discostarsi dal loro valore fondamentale anche in assenza di nuove informazioni rilevanti.

I sostenitori della *Behavioral Finance* ritengono che le fluttuazioni dei prezzi azionari possano essere legate ad un problema di aspettative degli investitori circa il valore futuro delle azioni, o addirittura ad un eccesso di ottimismo o pessimismo conseguente all'ascolto di una serie di buone o cattive notizie sul futuro.

Se un investitore ritiene che un'azione in futuro possa valere molto, oggi sarà disposto ad acquistarla ad un prezzo piuttosto alto nella convinzione di poterla rivendere l'anno successivo ad un prezzo ancora superiore; le azioni possono quindi aumentare di prezzo solo perché l'investitore si aspetta che ciò accada. Cosa succede dopo un certo periodo? Questo lievitare dei prezzi, non essendo supportato da alcun dato effettivo, ma essendo determinato solo dall'entusiasmo degli investitori, è destinato a spegnersi, e quindi si osserverà un crollo con una repentina discesa dei prezzi[11].

Questo fenomeno prende il nome di bolla speculativa e se hanno prove empiriche risalenti addirittura diciassettesimo secolo come descritto da Mackey nel suo libro "La pazzia delle folle" pubblicato nel 1841, nel quale descrive la "bolla dei tulipani" che si diffuse in Olanda nel 1634, quando i bulbi di tulipano arrivarono a costare più dell'oro[12].

---

[9] Scienza che studia i processi psichici, coscienti e inconsci, cognitivi (percezione, attenzione, memoria, linguaggio, pensiero ecc.) e dinamici (emozioni, motivazioni, personalità ecc.). *Dizionario Treccani*

[10] Scienza e disciplina che ha per oggetto i fenomeni sociali indagati nelle loro cause e manifestazioni, nei loro processi ed effetti, nei loro rapporti reciproci e con altri fenomeni. *Dizionario Treccani*

[11] Barone, R. (2003). From efficient markets to behavioral finance. *Available at SSRN 493545*.

[12] *Mackay C. La pazzia delle folle Il Sole 24ore 2000:* Egli narra che nel giro di pochi anni i bulbi di tulipano iniziarono ad essere molto ricercati, soprattutto in Olanda, prima dai nobili e pian piano anche dalle classi medie. Nel 1634 la smania di possedere questi tulipani era tale da spingere gli Olandesi a trascurare le proprie attività, per dedicarsi al commercio del tulipano. Pare che in Olanda ci fossero solo due radici, di cui una era posseduta da un ricco mercante. Quest'ultima finì nello stomaco di un povero marinaio che, ignaro dell'importanza che il tulipano stava avendo in quell'epoca, pensò bene di utilizzarla per condire una aringa. Il povero marinaio capì troppo tardi di aver ingoiato un tesoro! La domanda di tulipani crebbe così tanto nel 1636 che vennero istituiti mercati regolari per la loro vendita alla Borsa Valori di Amsterdam, Rotterdam, etc. Gli speculatori, avendo massima fiducia in questo mercato, iniziarono a commerciare in tulipani, acquistando quando il prezzo scendeva e vendendo quando saliva. Tutti pensavano che questa mania per i tulipani sarebbe durata in eterno. Ma fortunatamente alcuni più prudenti si resero conto che non era così. I ricchi non compravano più tulipani, ma li vendevano per guadagnare il 100%. Iniziò a diffondersi uno stato di sfiducia e pian piano il valore dei tulipani crollò definitivamente.

In generale possiamo affermare che le critiche elaborate da coloro che confutano l'ipotesi che i mercati siano efficienti si articolano in:

- **Critica all'ipotesi di razionalità**: secondo tale ipotesi appare irrealistico che tutti gli investitori che operano all'interno del mercato siano razionali nelle loro scelte di investimento, in quanto essi prenderanno le proprie decisioni sulla base di ragionamenti che deviano dal processo decisionale razionale. Inoltre, come abbiamo visto in precedenza, è proprio su questa critica che si basa la teoria della finanza comportamentale.
  L'economista statunitense Fischer Black[13] , in un suo articolo pubblicato nel 1986, conferma l'irrazionalità degli investitori attraverso l'esistenza dei *noise traders*; individui che assumono le loro decisioni di investimento senza effettuare nessuna analisi tecnica o fondamentalmente di un determinato titolo e quindi in maniera totalmente casuale ed irrazionale. Peraltro, anche se i *noise traders* rappresentassero solo una minoranza rispetto al totale degli investitori, le loro decisioni possono comunque provocare distorsioni nel mercato.
  Ulteriori studi sono stati svolti dagli economisti Kahneman e Riepe[14] nel 1988 i quali sostengono che esistono delle occasioni in cui gli investitori tendono a deviare dal processo razionale. Queste deviazioni possono essere definite in tre categorie: l'attitudine per il rischio, la diversa formazione delle aspettative e la sensibilità delle decisioni alla struttura del problema, che non permettono di seguire un percorso razionale. Un'ulteriore osservazione che fanno i due studiosi è che sul mercato non operano soltanto investitori privati, ma anche professionisti delle istituzioni finanziarie che sono soggetti ad ulteriori distorsioni in quanto gestiscono il denaro altrui e hanno obiettivi aziendali da raggiungere.

- **Critica all'ipotesi di arbitraggio**: ricordiamo che, secondo l'ipotesi dei mercati efficienti, i prezzi dei titoli riflettono completamente le informazioni disponibili e, se anche gli investitori irrazionali provocassero un cambiamento dei prezzi, interverrebbero gli arbitraggisti con il compito di riportare in equilibrio i prezzi annullando così ogni tipo di variazione.
  La critica all'ipotesi di arbitraggio è stata avanzata in quanto la ricerca empirica ha evidenziato che le operazioni di arbitraggio nei mercati finanziari non sono prive di rischio, come invece sosteneva L'EMH theory; è proprio in virtù della rischiosità delle strategie di

[13] BLACK F. (1986). *Noise*. The Journal of Finance, Vol. 41, No. 3, Papers and Proceedings of the Forty-Fourth Annual Meeting of the American Finance Association.
[14] KAHNEMAN D., Riepe. (1998). *The Psychology of non-Professional Investor*. Journal of Portfolio Management, Vol. 24, No.4

arbitraggio che i prezzi dei titoli rimangono distanti dai loro valori fondamentali per lunghi periodi. L'arbitraggio, quindi, non svolge la sua funzione di correzione dei prezzi e non riesce a garantire l'efficienza del mercato. Gli economisti Barberis e Thaler[15], in un loro articolo pubblicato nel 2003, hanno approfondito le tesi a sostegno di questa critica individuando quattro categorie di rischi legati all'arbitraggio:

1. Rischio fondamentale: si ha quando la pubblicazione di nuove informazioni fa calare il prezzo di un titolo anche se quest'ultimo fosse già sottostimato. Per difendersi da tale rischio gli arbitraggisti attuano delle strategie di *hedging*[16] negoziando titoli che sono considerati sostituti del precedente. Tuttavia, il rischio fondamentale non può essere del tutto eliminato in quanto, all'interno dei mercati di capitali, i titoli sostituti sono difficilmente perfetti.

2. *Noise trader risk*: questa tipologia di rischio è associata alla presenza dei cosiddetti *noise trader*; operatori del mercato che, non seguendo la razionalità economica, prendono decisioni di investimento che possono deviare significativamente dal valore "reale" di un asset. *Il noise trader risk* può portare a fluttuazioni dei prezzi che non riflettono le condizioni effettive del mercato o il valore intrinseco dell'asset. Queste fluttuazioni possono portare ad un rischio aggiuntivo per gli investitori che cercano di prendere decisioni di investimento razionali. Per superare tale problema e per non andare incontro a perdite, gli arbitraggisti attendono finché il prezzo dei titoli non si riallinei al suo valore fondamentale e ciò avviene vendendo titoli anche ad un prezzo inferiore rispetto a quello di acquisto.

3. Costi di implementazione: un'altra considerazione sfuggita ai promotori della teoria dei mercati efficienti è l'esistenza dei costi di transazione[17] i quali possono limitare i profitti degli arbitraggisti. Le operazioni tipiche di questi ultimi, come la

[15] Barberis, N. & Thaler, R. (2003). A survey of behavioral finance. *Handbook of the Economics of Finance*, *1*, 1053-1128.
[16] Strategia utilizzata per ridurre il rischio di perdita di un investimento aprendo una posizione in un asset correlato o in un mercato diverso, in modo tale da compensare eventuali perdite nell'investimento originale.
[17] Tipologia di costi relativa ad attività di scambio fra soggetti economici. In particolare, ciò che si rende necessario nell'attività di scambio è l'organizzazione e la diffusione di tutte quelle informazioni che conducono alla stipulazione specifica dei contratti e che costituiscono costi di transazione. *Dizionari Simone.*

compravendita[18] o la vendita allo scoperto[19], sono spesso soggette anche a restrizioni legali. Tutti i vari costi di implementazione complicano quindi il meccanismo di arbitraggio e lo rendono più rischioso.

4. Rischio di modello: è il rischio che il modello usato dagli arbitraggisti potrebbe non essere veritiero e che quindi non rappresenti correttamente la realtà. Tale rischio può portare a decisioni di investimento sbagliate o ad altre conseguenze negative, oltre a causare una forte incertezza nell'implementazione delle strategie di arbitraggio.

Questi quattro fattori di rischio, uniti ad altri elementi che si realizzano nella realtà empirica, concorrono a porre un forte limite all'arbitraggio.
"L'inesistenza di un arbitraggio del tutto efficace rende impossibile la correzione delle distorsioni del mercato, che risulta, dunque, inefficiente. La teoria dell'arbitraggio limitato dimostra che se da una parte gli investitori irrazionali causano deviazioni del prezzo dal valore fondamentale di un titolo, dall'altra gli investitori razionali spesso non hanno il potere di correggere tali scostamenti".[20]

Nel corso della storia si sono verificati molti eventi difficilmente compatibili con la teoria dei mercati efficienti, le cosiddette anomalie di mercato. Con questo termine si intendono tutte quelle situazioni in cui il prezzo di mercato di un bene o di un'attività finanziaria non riflette completamente le informazioni disponibili, generando inefficienze o discrepanze tra il prezzo effettivo e quello che sarebbe giustificato dalla teoria economica.
Le anomalie di mercato possono essere sfruttate dai trader più attenti per generare profitti, ma possono anche rappresentare una minaccia per l'efficienza dei mercati finanziari. L'EMH theory sostiene che i mercati finanziari dovrebbero essere efficienti e riflettere completamente tutte le informazioni disponibili, ma le anomalie di mercato suggeriscono che questo non sempre avviene.
Lo studioso Schwert[21] fu il primo a notare che quando le anomalie venivano scoperte e documentate, esse tendevano a scomparire o ad attenuarsi; proprio per questo si dice che esse siano più apparenti che reali. Tuttavia, è logico pensare che una volta scoperte, le anomalie vengano sfruttate nelle

[18] La compravendita (o vendita) è il contratto avente per oggetto il trasferimento della proprietà di una cosa o il trasferimento di un altro diritto verso il corrispettivo di un prezzo. *Articolo 1470 Codice Civile.*
[19] Operazione finanziaria che consiste nella vendita di strumenti finanziari non posseduti con successivo riacquisto e che appunto si effettua se si ritiene che il prezzo al quale gli strumenti finanziari si riacquisteranno sarà inferiore al prezzo inizialmente incassato attraverso la vendita. *Il Sole 24 Ore, Cosa sono le vendite allo scoperto e perché vietarle (non sempre) funziona, Cellino M. 2020*
[20] ASSONEBB (Associazione Nazionale Enciclopedia della Banca e della Borsa) *Critiche all'efficienza di mercato, Bankpedia.org*
[21] Schwert, G. W. (2003). Anomalies and market efficiency. *Handbook of the Economics of Finance*, *1*.

strategie di investimento degli agenti del mercato in modo da trarne vantaggio in termini di profitto, il che potrebbe causare la loro scomparsa.
Le anomalie di mercato più diffuse sono:

- **Effetto gennaio**: gli economisti Rozeff e Kinney[22], in un articolo pubblicato nel 1976, furono i primi a documentare un comportamento anomalo dei rendimenti del mercato in alcuni periodi dell'anno ed in particolare un rendimento medio più alto nel mese di gennaio, rispetto agli altri mesi. I due studiosi presero in esame le azioni del NYSE tra il 1904 ed il 1974 ed osservarono che il rendimento medio nel mese di gennaio era pari al 3.48%, chiaramente più alto dello 0.42% osservato negli altri mesi. L'ipotesi più accreditata per spiegare questo fenomeno è quella che vede come causa un generale bilanciamento dei titoli nel mese di dicembre. Gli investitori sono spinti a vendere i titoli poco performanti a fine anno per detrarre le minusvalenze dall'imponibile da tassare e di conseguenza, con l'abbassamento del prezzo, questi diventano appetibili. Tuttavia, gli operatori aspetteranno il mese di gennaio per acquistarli per evitare che vengano inclusi nello stesso anno fiscale.

- **Effetto weekend**: noto anche come "effetto lunedì", indica la tendenza dei prezzi dei titoli a chiudere in ribasso il lunedì rispetto al venerdì precedente. Tale anomalia è stata oggetto di studio dell'economista Kenneth French[23] il quale, analizzando i rendimenti di alcune azioni tra il 1953 ed il 1977, ha scoperto che i rendimenti tendono ad essere negativi di lunedì e generalmente positivi negli altri giorni della settimana.

Average percent return from the close of the previous trading day to the close of the day indicated [a]

	Monday	Tuesday	Wednesday	Thursday	Friday
1953	−0 2488	−0 0570	0 1181	0 0641	0 0110
1954	0 0362	0 0260	0 1746	0 1959	0 2524
1955	−0 2351	0 0857	0 2497	0 0020	0 3135
1956	−0 1445	−0 0393	−0 0649	0 0327	0 2069
1957	−0 5102	−0 0560	0 3083	−0 0237	−0 0949
1958	0 0301	0 0830	0 1166	0 1246	0 2043
1959	−0 1403	0 0865	0 0066	0 0485	0 1819
1960	−0 3487	0 0121	0 0286	0 0560	0 1604
1961	−0 0620	0 0440	0 2011	0 0631	0 1311
1962	−0 3263	0 0388	0 0404	0 0343	−0 1070
1963	−0 0836	0 1248	0 0525	0 0588	0 0969
1964	−0 0400	−0 0463	0 1023	0 0585	0 1692
1965	−0 1286	0 0505	0 0740	0 0354	0 1512
1966	−0 2645	−0 0414	0 1416	−0 1049	−0 0064
1967	−0 1755	0 1062	0 1343	0 2142	0 1026
1968	0 0007	0 0623	0 2410	−0 0664	0 0086
1969	−0 3503	−0 0691	0 0754	0 0404	0 0842
1970	−0 2790	−0 1230	0 2677	−0 0361	0 1370
1971	−0 0621	0 0872	0 0489	−0 0193	0 0899
1972	−0 1529	0 0206	0 1469	0 0501	0 1935
1973	−0 4738	0 0338	−0 0578	0 1293	−0 0877
1974	−0 3784	0 1677	−0 1015	−0 0956	−0 2676
1975	0 1918	−0 2279	0 1450	0 2250	0 2383
1976	0 1089	0 1496	0 1483	−0 0433	−0 0275
1977	−0 1274	−0 1126	−0 1091	0 0237	0 0403

[a] Returns for periods including a holiday are omitted. These returns are defined as $R_t = \ln(P_t/P_{t-1})\ 100$

Fig.2 – French, K. R. (1980)

[22] Rozeff, M. S., & Kinney Jr, W. R. (1976). Capital market seasonality: The case of stock returns. *Journal of financial economics*, *3*(4).
[23] French, K. R. (1980). Stock returns and the weekend effect. *Journal of financial economics*, *8*(1).

I sostenitori della finanza comportamentale ipotizzano che tale effetto sia causato dalla negatività che circonda una nuova settimana lavorativa; altri invece spiegano tale fenomeno col fatto che molte aziende tendono a pubblicare le notizie "cattive" il venerdì sera, con la conseguenza che molti trader venderanno i loro titoli proprio in quel giorno facendone calare il prezzo alla riapertura del mercato il lunedì successivo.

- **Effetto cambio del mese**: in alcuni studi è emerso un aumento della redditività dei titoli azionari durante l'ultimo giorno di negoziazione del mese e, in alcuni casi, anche fino ai primi tre giorni del mese successivo. Il primo studioso che analizzò questo effetto fu Ariel[24] nel 1987 che prese in considerazione il mercato statunitense. Altra prova empirica di questa anomalia si ha con lo studio condotto da Barone[25] in Italia nel 1990 nel quale, come è possibile notare dalla figura sottostante, dimostrò la veridicità di questo fenomeno.

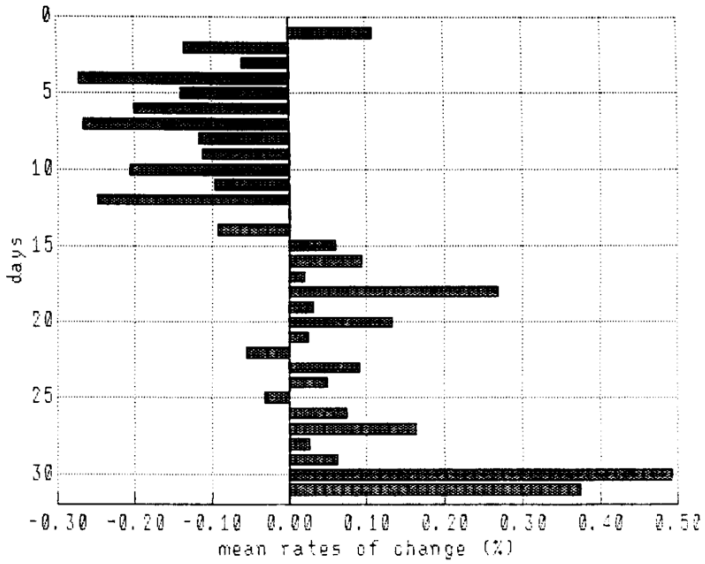


Fig.3 – Barone, E. (1990)

La spiegazione più plausibile per questo fenomeno è il pagamento dei salari, che avvenendo verso la fine del mese, condizionerebbe l'aumento della domanda di titoli.

- **Halloween indicator**: indicata anche con l'espressione "sell in May and go away", questa anomalia prevede un forte effetto stagionale secondo cui i rendimenti delle azioni dovrebbero essere più elevati nel periodo compreso tra novembre e aprile rispetto al periodo dell'anno

[24] Ariel, R. A. (1987). A monthly effect in stock returns. *Journal of financial economics*, *18*(1), 161-174.
[25] Barone, E. (1990). The Italian stock market: efficiency and calendar anomalies. *Journal of Banking & Finance*, *14*(2-3).

compreso tra maggio e ottobre. Di questa anomalia ne hanno trovato riscontro empirico gli economisti Bouman e Jacobsen[26] in ben 36 mercati sui 37 analizzati.

- **Impatto post-trimestrali**: tale anomalia si riferisce agli effetti che i risultati finanziari trimestrali di un'azienda possono avere sul mercato azionario e sull'opinione degli investitori. Dopo la pubblicazione dei risultati trimestrali, gli investitori possono reagire in modo positivo o negativo a seconda delle prestazioni dell'azienda rispetto alle aspettative. Se l'azienda ha superato le previsioni, il prezzo delle sue azioni potrebbe aumentare e gli investitori potrebbero avere un'opinione più positiva sulla società e saranno spinti ad acquistare ancora più azioni. Al contrario, se l'azienda ha ottenuto risultati inferiori alle aspettative, il prezzo delle sue azioni potrebbe diminuire e gli investitori potrebbero avere un'opinione più negativa sulla società.

- **Effetto vacanza**: questa anomalia descrive la tendenza del mercato azionario a guadagnare nell'ultimo giorno di negoziazione prima di una festività. Tale fenomeno può essere causato dall'ottimismo e dalla positività che investe le persone in questi giorni. Sempre Barone[27], nello studio condotto in Italia nel 1990, ha scoperto che le variazioni dei prezzi nei giorni festivi sono positive nel 60% dei casi, contro il 49% degli altri giorni dell'anno.

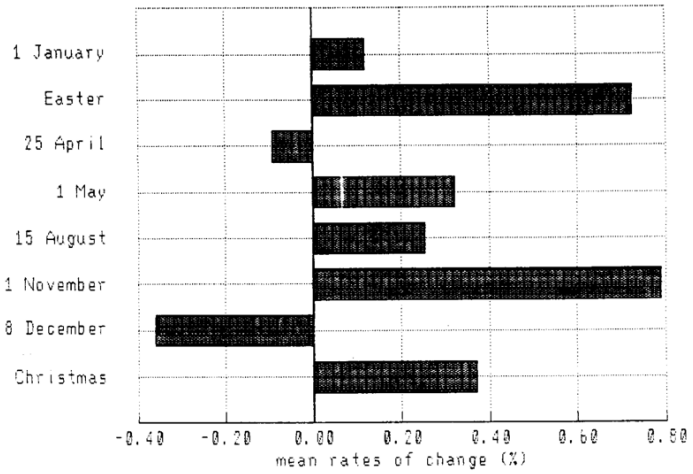


Fig.4 – Barone, E. (1990)

- **Effetto momentum**: questo effetto si riferisce alla tendenza di un titolo che ha mostrato un andamento positivo in passato di continuare a performare, e viceversa per un titolo che ha

[26] Bouman, S., & Jacobsen, B. (2002). The Halloween indicator,"sell in May and go away": Another puzzle. *American Economic Review*, *92*(5), 1618-1635.
[27] Barone, E. (1990). The Italian stock market: efficiency and calendar anomalies. *Journal of Banking & Finance*, *14*(2-3).

mostrato una performance negativa in passato. Questa anomalia è causata dal fatto che gli investitori tendono a reagire esageratamente alle informazioni e quindi i titoli non assorbono immediatamente le nuove informazioni nel prezzo, ma lo fanno in modo più graduale.

1.3 IL RUOLO DEL SENTIMENT

Dopo aver analizzato la teoria dei mercati efficienti e le rispettive critiche, si introdurrà l'obiettivo del presente lavoro. La domanda principale alla quale vuole rispondere il mio lavoro è capire se è possibile prevedere il rendimento di un determinato titolo conoscendo l'opinione generale della comunità nei confronti di quella determinata azienda.

Le prime ricerche sulla previsione del mercato azionario si basavano sulla teoria del *random walk* e sull'ipotesi dei mercati efficienti. Secondo l'EMH, come abbiamo visto, i prezzi del mercato azionario sono in gran parte guidati dalle nuove informazioni, piuttosto che da prezzi presenti e passati. Poiché le notizie sono imprevedibili, i prezzi del mercato azionario seguiranno uno schema casuale e non possono essere previsti con una precisione superiore al 50%.

Ci sono due problemi con questa teoria. In primo luogo, numerosi studi dimostrano che i prezzi del mercato azionario non seguono un andamento casuale e possono effettivamente essere previsti mettendo così in discussione le ipotesi di base dell'EMH. In secondo luogo, ricerche recenti suggeriscono che le notizie possono essere imprevedibili, ma anche che possono essere estratti dai social media online (blog, feed di Twitter, ecc.) indicatori molto interessanti per prevedere i cambiamenti del mercato azionario e dei vari indicatori economici e commerciali[28].

Nello specifico, per predire gli andamenti dei titoli, gli studiosi si sono concentrati sullo studiare come l'umore pubblico influenza i rendimenti.

La *sentiment analysis* è una tecnica di elaborazione del linguaggio naturale volta a fotografare l'umore della collettività in un determinato momento rispetto ad un determinato fenomeno di interesse.

Questa analisi è utile in molte applicazioni, come il monitoraggio dei social media, le analisi dei *feedback* dei clienti, la valutazione della reputazione online di un'azienda, le ricerche di marketing e molte altre ancora. Grazie a questa tecnica, è possibile analizzare grandi quantità di dati testuali e ottenere informazioni utili per prendere decisioni informate.

Per effettuare l'analisi del *sentiment* si prendono come base di studio i commenti pubblici attinenti ad un determinato argomento in un arco temporale prestabilito. In una prima fase vengono filtrati per parole chiave in modo da ottenere un set di dati strettamente correlato al fenomeno di interesse. Questi verranno poi analizzati grazie a delle tecniche di *data mining* che permettono sia di classificarli

[28] Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of computational science*, 2(1), 1-8.

secondo una polarità positiva o negativa, sia di classificarli in base allo stato d'animo preciso che prova l'autore del commento (felicità, rabbia, calma, sicurezza, pessimismo, ecc.).

Occorre precisare che un software efficace con cui praticare *sentiment analysis* deve essere in grado di comprendere efficacemente il significato di ogni singola parola, inclusi i casi in cui ce ne sia più di uno (polisemia).

I settori in cui è possibile utilizzare questa analisi sono molteplici: dalla politica ai mercati azionari, dal marketing alla comunicazione, dall'ambito sportivo a quello delle scienze mediche e naturali. Ma non solo: è possibile anche misurare le preferenze del consumatore in relazione a programmi televisivi, film e spettacoli di vario genere. Per esempio, la *sentiment analysis* è stata utilizzata per mostrare come l'attività di chat online può prevedere le vendite di libri[29] e come l'analisi dei blog online può prevedere le vendite dei film[30].

Elemento imprescindibile ed alla base di queste analisi è il *text mining*, oggetto di analisi del prossimo capitolo.

[29] Gruhl, D, Guha, R, Kumar, R, Novak, J, & Tomkins, A. (2005) *The predictive power of online chatter*. (ACM, New York, NY, USA), pp. 78–87.

[30] Mishne, G & Glance, N. (2006) *Predicting Movie Sales from Blogger Sentiment*. AAAI 2006 Spring Symposium on Computational Approaches to Analysing Weblogs

CAPITOLO SECONDO

TEXT MINING

2.1 INTRODUZIONE AL TEXT MINING

Al giorno d'oggi tutte le aziende più importanti del panorama globale, oltre ovviamente ai governi, si servono dei *big data* per analizzare e scovare *trend* nascosti al fine di prendere le decisioni migliori. Con il termine "*big data*" si fa riferimento a grandi quantità di dati che sono troppo complessi e vasti per essere gestiti e analizzati con le tecniche tradizionali di gestione dei dati. Questi dati possono provenire da innumerevoli fonti, come ad esempio transazioni finanziarie, sensori, social media, dati geospaziali, e altro ancora. La caratteristica principale dei *big data* è la loro dimensione, che può arrivare all'ordine dei petabyte o addirittura exabyte[31]. L'obiettivo dell'analisi dei *big data* è quello di estrarre informazioni utili e *insights* per supportare le decisioni aziendali, la previsione, l'ottimizzazione dei processi, la personalizzazione dei prodotti e servizi, e altro ancora.
Esistono molte tipologie di *big data*, ma le due macrocategorie sono:

- Dati strutturati: sono principalmente dati di tipo numerico e sono organizzati in tabelle. I dati strutturati sono relativamente facili da gestire e analizzare utilizzando le tecniche tradizionali di gestione dei dati, come l'utilizzo di *query* SQL. Questa organizzazione dei dati facilita l'accesso alle informazioni specifiche e consente di identificare rapidamente i dati che sono importanti per l'analisi.
- Dati non strutturati: sono quei dati che non hanno una struttura organizzata, ovvero non sono organizzati in modo tabellare con righe e colonne, ma sono invece rappresentati in modo libero e flessibile, come ad esempio immagini, audio, video, testo libero, e-mail, messaggi sui *social media* e altro ancora.

Come facilmente intuibile, i dati strutturati sono molto più semplici da gestire e analizzare rispetto ai dati non strutturati. Tuttavia, questi ultimi sono presenti in quantità nettamente superiore e proprio la grande difficoltà nel gestirli permette, a chi ci riesce, di ottenere un enorme vantaggio competitivo.

[31] $1PB = 10^6GB\ ;\ 1EB = 10^9GB$

Per procedere allo studio della *sentiment analysis* è fondamentale riuscire ad estrapolare il maggior numero di informazioni possibile dai dati non strutturati di tipo testuale. Ciò è possibile grazie al *text mining*.

Il *text mining* è una tecnica di analisi dei dati che consente di estrarre informazioni utili da grandi quantità di testo non strutturato utilizzando algoritmi di elaborazione del linguaggio naturale (*Natural Language Processing*, NLP).

Questo metodo di analisi si sta via via diffondendo molto rapidamente, trovando sempre più campi applicativi. Tra i principali troviamo:

- *Customer Opinion Survey*: consiste nell'analisi automatica delle segnalazioni e dei reclami pervenuti su prodotti o servizi offerti da un'azienda, per aiutare quest'ultima a capire meglio i bisogni dei propri clienti e adottare strategie di marketing più efficaci.
- Monitoraggio dei *social media*: il *text mining* può essere utilizzato per monitorare e analizzare i post sui social media, consentendo alle aziende di comprendere meglio l'opinione pubblica sui loro prodotti o servizi e di identificare le tendenze emergenti.
- Analisi dei documenti giuridici: questa analisi può essere utilizzata per analizzare contratti o sentenze, per estrarre informazioni pertinenti e individuare eventuali rischi legali.
- Analisi dei dati biomedici: il *text mining* può essere utilizzato per analizzare i dati biomedici, come i rapporti di studi clinici, per identificare nuovi farmaci o trattamenti medici.
- Analisi dei dati finanziari: consiste nell'analisi di bilanci o report trimestrali, per aiutare le aziende ad individuare tendenze e identificare eventuali rischi.

Il processo di *text mining* varia in base all'obiettivo della ricerca che si intende sviluppare. Nel nostro caso, quello volto all'analisi del *sentiment* può essere sintetizzato in tre fasi:

a) *Pre-processing* dei testi: consiste nel reperimento dei dati da varie fonti (web, giornali, social network ecc.) e nella loro formattazione.
b) *Lexical processing*: in questa fase prevale l'uso della linguista e del *Natural Language Processing* (NLP). Nello specifico, questa fase consiste dapprima nel "pulire" i testi eliminando simboli di punteggiatura, *stop words*[32], e altre informazioni indesiderate; in seguito, il testo viene *tokenizzato,* cioè suddiviso in unità di significato più piccole, dette

[32] Parole che vengono considerate poco significative nell'ambito delle analisi testuali poiché usate molto frequentemente nel linguaggio quotidiano. Queste hanno il solo scopo di collegare i vari elementi di una frase come articoli, preposizioni, congiunzioni e pronomi.

token, ed infine attraverso un'analisi lessicale si attribuisce un significato ad ogni *token* e si ricercano le relazioni tra di essi.

c) Analisi dei dati utilizzando tecniche di *machine learning* al fine di estrarre tutte le informazioni utili, per poi rappresentarle in vari formati come grafici, tabelle, diagrammi, ecc. per aiutare a comprendere meglio i risultati e trarne conclusioni.

Si evince che il processo più importante è il secondo; quello in cui, tramite le tecniche del NLP, si manipola il testo e si riesce a dare un peso ed un significato ad ogni parola per poter poi trovare delle relazioni nascoste.

Data l'importanza che riveste il *Natural Language Processing* all'interno dell'analisi del testo, procederemo nel prossimo paragrafo ad approfondire questo tema analizzando nello specifico il suo funzionamento.

2.2 THE NATURAL LANGUAGE PROCESSING

Il *text mining* e il *natural language processing* (NLP) sono due discipline strettamente correlate nell'ambito dell'elaborazione del linguaggio naturale. Tuttavia, queste spesso si sovrappongono, in quanto il *text mining* può utilizzare tecniche di NLP per la comprensione del testo e la classificazione dei dati; procederemo quindi a definire queste due discipline per comprendere al meglio i rispettivi campi di ricerca.

Il *text mining* si concentra sull'elaborazione automatica dei testi per estrarre informazioni utili, come ad esempio la scoperta di *pattern*, tendenze, associazioni e altre relazioni tra i dati. Per fare ciò, il *text mining* utilizza tecniche di analisi statistiche e di *machine learning*.

Il *natural language processing*, invece, si concentra sulla comprensione del linguaggio naturale e sulla sua elaborazione automatizzata. L'obiettivo principale del NLP è quello di far comprendere ai computer il significato dei testi e di farli in grado di elaborare le richieste in linguaggio naturale.

Al giorno d'oggi questa tecnica di elaborazione del linguaggio viene usata non soltanto per comprendere i testi, ma anche per il riconoscimento vocale. Gli esempi più lampanti sono gli assistenti virtuali come Siri, sviluppato da Apple, e Alexa sviluppato invece da Amazon.

La storia del *natural language processing* è stata caratterizzata da numerosi sviluppi e innovazioni tecnologiche. Il primo sistema di elaborazione del linguaggio naturale moderno è stato sviluppato solo negli anni '50, quando il matematico e informatico britannico Alan Turing propose il "test di Turing"[33] per valutare la capacità di un computer di eseguire una conversazione naturale.

[33] Il test di Turing è un gioco in cui una persona e un computer cercano di convincere un giudice che sono entrambi persone. Se il giudice non riesce a distinguere il computer dalla persona, allora si può dire che il computer ha superato il

Negli anni ‘60 e ‘70, il NLP ha fatto grandi progressi grazie allo sviluppo di algoritmi di analisi semantica e sintattica, che hanno permesso ai computer di comprendere e generare frasi complesse in linguaggio naturale. Lo studio più importante in questo ambito fu quello svolto dall’informatico e professore tedesco Joseph Weizenbaum, il quale, nel 1966, pubblicò un articolo nel quale descrisse il programma “ELIZA” che aveva appena sviluppato. ELIZA simulava una conversazione tra un utente ed un computer imitando le risposte di uno psicoterapeuta ed utilizzando tecniche di elaborazione del linguaggio naturale. Un esempio del suo funzionamento può essere preso dallo studio pubblicato da Weizenbaum[34]:

```
User:   I am unhappy.
ELIZA:  DO YOU THINK COMING HERE WILL HELP YOU NOT TO BE UNHAPPY
User:   I need some help, that much seems certain.
ELIZA:  WHAT WOULD IT MEAN TO YOU IF YOU GOT SOME HELP
User:   Perhaps I could learn to get along with my mother.
ELIZA:  TELL ME MORE ABOUT YOUR FAMILY
User:   My mother takes care of me.
ELIZA:  WHO ELSE IN YOU FAMILY TAKES CARE OF YOU
User:   My father.
ELIZA:  YOUR FATHER
User:   You are like my father in some ways.
```

Fig.5 - Weizenbaum, J. (1966)

Quando nel 1976[35] l’informatico tedesco spiegò il funzionamento di questo programma, il pubblico rimase molto sorpreso.

ELIZA non comprende realmente il significato delle parole dell’utente, ma è un programma sorprendentemente semplice che utilizza il *pattern matching* per riconoscere frasi come "Ho bisogno di X" e tradurle in output adeguati come "Cosa significherebbe per te l’ottenimento di X?".

L’imitazione della conversazione umana da parte di ELIZA ebbe un notevole successo: molte persone che interagirono con questo programma arrivarono a credere che quest’ultimo capisse davvero loro ed i loro problemi anche dopo averne scoperto il suo funzionamento.

Ovviamente i moderni *chatbot* sono molto più sofisticati di ELIZA il cui funzionamento però è estremamente utile per comprendere le basi dell’elaborazione del linguaggio naturale.

Procediamo ora analizzando i vari passaggi utili alla manipolazione del testo ed all’estrazione di dati da esso.

---

test e dimostrato di avere un comportamento intelligente simile a quello umano. In sostanza, il test di Turing è un modo per verificare se un computer può “pensare” come un essere umano.

[34] Weizenbaum, J. (1966). ELIZA—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, *9*(1), 36-45.

[35] Weizenbaum, J. (1976). Computer power and human reason: From judgment to calculation.

Inizieremo illustrando lo strumento più importante per descrivere i modelli di testo: la *regular expression*. Le espressioni regolari sono sequenze di caratteri che possono essere utilizzate per specificare le stringhe che si desiderano estrarre da un documento.
Passeremo poi ad analizzare una serie di operazioni chiamate collettivamente *text normalization*. Normalizzare il testo significa convertirlo in una forma più comoda per analizzarlo e l'operazione principale è rappresentata dalla *tokenization*. Per tokenizzazione del testo si intende la sua suddivisone in singole parole chiamate appunto *token*. Questa operazione può sembrare banale in quanto ogni parola è solitamente preceduta e successa da uno spazio, ma esistono alcune parole soprattutto straniere, come "*New York*" e "*rock 'n' roll*", che devono essere trattate come se fossero un'unica parola nonostante contengano spazi. Questa operazione è particolarmente utile nell'analisi di testi scritti in lingue, come il giapponese, che non utilizzano spazi per separare le parole.
Inoltre, per l'elaborazione dei testi presenti sui *social media*, è necessario riuscire a tokenizzare anche le *emoticon* e gli *hashtag*.
Un'altra parte della normalizzazione del testo è la *lemmatization*, che consiste nel ridurre ogni parola alla sua forma base, detta lemma. In altre parole, il lemma è la forma di una parola che rappresenta il significato di base della parola stessa. Ad esempio, il lemma del verbo "correndo" è "correre". Questa tecnica è particolarmente importante in quanto consente di normalizzare il testo e semplificare la comparazione e l'analisi delle parole, oltre ad essere essenziale per elaborare lingue morfologicamente complesse. Lo *stemming*, invece, può essere definito come una versione semplificata della lemmatizzazione, in cui ci si limita a eliminare i suffissi dalla fine di ogni parola.
La normalizzazione del testo comprende anche la *sentence segmentation*: la suddivisione di un testo in singole frasi, utilizzando indicazioni come punti o esclamazioni.
Infine, introdurremo una metrica chiamata *edit distance* che misura la somiglianza tra due stringhe in base al numero di modifiche (inserimenti, cancellazioni, sostituzioni) necessarie per trasformare una stringa nell'altra.
L'*edit distance* è un algoritmo che trova applicazione in tutti i processi linguistici, dalla correzione ortografica fino al riconoscimento vocale.
Nei sottoparagrafi seguenti procederemo all'analisi di ogni operazione accennata in precedenza effettuando anche esempi con comandi UNIX.

2.2.1 REGULAR EXPRESSIONS

Le espressioni regolari, spesso abbreviate in "*regex*", sono una tecnica utilizzata nell'elaborazione del linguaggio naturale e nella manipolazione di stringhe di testo. Formalmente una *regular expression* è una sequenza di caratteri che definisce un *pattern*, ovvero una combinazione di caratteri che descrive una stringa.

Questa tecnica è particolarmente utile per ricercare un determinato *pattern* in un *corpus*. Per *corpus* si intende il documento o il set di documenti in cui si intende effettuare la ricerca.
L'operazione che permette di unire vari caratteri per formare un *pattern* è detta concatenazione. Se ricerchiamo una tale parola ci verranno mostrate tutte le stringhe del *corpus* che la conterranno.

Regex	Example Patterns Matched
/woodchucks/	"interesting links to woodchucks and lemurs"
/a/	"Mary Ann stopped by Mona's"
/!/	"You've left the burglar behind again!" said Nori

Fig.6 - Jurafsky D. & Martin J.H. (2023). Speech and Language Processing. Third edition.

È importante notare che le *regular expressions* sono *case sensitive*: distinguono cioè tra lettere maiuscole e minuscole. Per esempio, se cerchiamo la parola "stato" non ci verranno mostrate le stringhe contenenti il *pattern* "Stato". Per ovviare a questo problema introduciamo il concetto di disgiunzione: tramite questo metodo, che si applica utilizzando le parentesi quadre, chiediamo al sistema di ricercare un carattere o un altro. Nell'esempio precedente, per ottenere le corrispondenze della parola "stato" indipendentemente se essa sia scritta con lettera maiuscola o minuscola, dovremmo effettuare la ricerca scrivendo "[sS]tato".

Regex	Match	Example Patterns
/[wW]oodchuck/	Woodchuck or woodchuck	"Woodchuck"
/[abc]/	'a', 'b', *or* 'c'	"In uomini, in soldati"
/[1234567890]/	any digit	"plenty of 7 to 5"

Fig.7 - Jurafsky D. & Martin J.H. (2023). Speech and Language Processing. Third edition.

Soffermandoci sull'ultimo esempio della tabella precedente, possiamo notare che è stata effettuata la ricerca di un qualsiasi numero. In casi come questo in cui esiste una sequenza ben definita associata a un insieme di caratteri, le parentesi possono essere usate con il trattino [-] per specificare un carattere qualsiasi in un intervallo.

Regex	Match	Example Patterns Matched
/[A-Z]/	an upper case letter	"we should call it 'Drenched Blossoms' "
/[a-z]/	a lower case letter	"my beans were impatient to be hoed!"
/[0-9]/	a single digit	"Chapter 1: Down the Rabbit Hole"

Fig.8 - Jurafsky D. & Martin J.H. (2023). Speech and Language Processing. Third edition.

Le parentesi quadre possono essere utilizzate anche per specificare ciò che un singolo carattere non può essere, utilizzando il simbolo "^". Se questo viene usato subito dopo aver aperto la parentesi quadra, lo schema risultante viene negato. Per esempio, lo schema /[ˆa]/ corrisponde a qualsiasi singolo carattere (compresi i caratteri speciali) tranne a.
Inoltre, è possibile usare un punto interrogativo dopo un carattere nel caso in cui si vogliono ricercare tutte le parole che contengono e che non contengono quel carattere.

Regex	Match	Example Patterns Matched
/woodchucks?/	woodchuck or woodchucks	"woodchuck"
/colou?r/	color or colour	"color"

Fig.9 - Jurafsky D. & Martin J.H. (2023). Speech and Language Processing. Third edition.

Il punto interrogativo sta a significare "zero o una sola istanza del carattere precedente". In altre parole, è un modo per specificare quante istanze di qualcosa vogliamo, cosa molto importante nelle espressioni regolari.
Se invece vogliamo ricercare "zero o più istanze del carattere precedente" dobbiamo utilizzare l'asterisco, chiamato in gergo "*Kleene* *". Per esempio /a*/ significa che vogliamo ricercare qualsiasi stringa che contenga zero o più "a". I risultati corrisponderanno ad "a" o "aaaaaa", ma anche ad una stringa che non contiene il carattere che stiamo ricercando come per esempio "ping pong". Per superare questo problema, e quindi escludere dai risultati le stringhe con non contengono il carattere in questione, dovremmo effettuare la ricerca /aa*/, che corrisponde ad individuare una "a" seguita da zero o più "a". Anche modelli più complessi possono essere ripetuti; per esempio /[ab]*/ significa "zero o più a o b" ed i possibili *patterns matched* possono essere stringhe come "aaaa", "ababab" o "bbbb".
Questa tecnica può essere utilizzata anche per ricercare più cifre, operazione utile per individuare per esempio i prezzi all'interno di un testo. Per effettuare questa operazione possiamo ricercare:
/[0-9][0-9]*/.
È chiaro che l'operazione sopra descritta è molto macchinosa e per i ricercatori può essere fastidioso dover scrivere per due volte la stessa cosa. Esiste infatti un modo più semplice, il "*Kleene* +", che permette appunto di ricercare "una o più occorrenze del carattere o dell'espressione immediatamente precedente". Pertanto, l'espressione scritta in precedenza può essere modificata in /[0-9]+/.
Altro carattere molto importante nella ricerca dei *pattern* è il punto. Questo viene definito "*wildcard*" in quanto viene abbinato dal sistema ad ogni carattere e può essere usato nel modo seguente: digitando /beg.n/ ci appariranno tutte le parole che iniziano per "beg" e terminano in "n".

Un ultimo operatore molto importante da considerare è il simbolo " | " che ci permette di inserire in una singola riga di comando più parole da ricercare: digitando quindi /day|night/ ci verranno restituite tutte le stringhe contenenti o la parola "*day*" o la parola "*night*".
Il secondo uso principale delle *regular expressions* è quello della sostituzione.
Per sostituire una parola con un'altra è innanzitutto indispensabile ricercare tale parola all'interno del testo utilizzando tutte le tecniche spiegate in precedenza. Nello specifico, tale operazione viene effettuata seguendo la seguente struttura: "s/(parola da sostituire)/parola nuova/".
È importante notare che la parola da sostituire deve sempre essere scritta tra parantesi; in caso contrario verrà sostituita con la nuova parola l'intera stringa contenente il *pattern* che abbiamo ricercato.
Altra operazione che può essere utile è quella di modificare una specifica sequenza di caratteri. Ipotizziamo di voler inserire delle parentesi graffe intorno ad ogni numero del testo trasformando per esempio 23 in {23}.
Per fare ciò possiamo scrivere il seguente codice:
s / ([0-9]+) / {\1} /
Dove l'operatore "\1" va a richiamare e quindi a copiare il contenuto delle parentesi tonde.

2.2.2 TEXT NORMALIZATION

Prima di effettuare una qualsiasi elaborazione di un testo in linguaggio naturale, quest'ultimo deve essere dapprima normalizzato, cioè, come spiegato in precedenza, convertito in una forma più semplice da analizzare. Il processo di normalizzazione consta di varie fasi che possono essere raggruppate come segue[36]:

1- Tokenizzazione delle parole
2- Normalizzazione dei formati delle parole
3- Segmentazione delle frasi

Nelle prossime sezioni esamineremo ciascuno di questi compiti.

Per tokenizzazione del testo si intende l'operazione di suddivisione del testo in singole parole. I software che ci aiutano ad effettuare questa operazione sono particolarmente complessi e devono prendere in considerazione un numero enorme di varianti. Per esempio, alcune volte potremmo voler considerare la punteggiatura come un token separato in quanto utile per comprendere al meglio la frase ed indentificarne i confini; altre volte invece ci farebbe comodo se i simboli di punteggiatura che si trovano all'interno di una parola vengano considerati come parte di essa, in esempi come

[36] Jurafsky D. & Martin J.H. (2023). Speech and Language Processing. Third edition.

m.p.h., Ph.D. e AT&T. Stesso discorso vale per i caratteri speciali e per i numeri che devono essere mantenuti nelle date (02/02/06) e nei prezzi (45,55 dollari); sarebbe infatti molto scomodo se il software segmentasse il prezzo in token separati di "45" e "55". Ci sono poi gli URL (https://www.stanford.edu), gli hashtag di Twitter (#nlproc) e gli indirizzi e-mail (someone@cs.colorado.edu). Tutto ciò per rendere l'idea di quanto un programma debba essere capace non soltanto di analizzare ogni carattere ma anche di comprendere il contesto in cui si trova.
Gli algoritmi di tokenizzazione più evoluti ed avanzati sono addirittura in grado di tokenizzare espressioni multi-parola, considerando per esempio "*rock'n'roll*" come unico token, e dividere le cosiddette "*clitic contractions*" come "*we'll*" in "*we will*".
Poiché la tokenizzazione è il primo passo per l'elaborazione del testo, questa deve essere eseguita in tempi molto rapidi. Il metodo standard per la tokenizzazione consiste quindi nell'utilizzare algoritmi deterministici basati su espressioni regolari compilate in automi a stati finiti molto efficienti. In seguito riporto un esempio di tokenizzazione che ho effettuato utilizzando la funzione "nltk.word_tokenize( )" del *Natural Language Toolkit* (NLTK) con il linguaggio di programmazione *Python*.

```
In [1]: import nltk

In [2]: sentence = 'That U.S.A. poster-print costs €18,90 on this website: www.cartoline.it '

In [3]: tokens = nltk.word_tokenize(sentence)

In [4]: tokens
Out[4]:
['That',
 'U.S.A.',
 'poster-print',
 'costs',
 '€18,90',
 'on',
 'this',
 'website',
 ':',
 'www.cartoline.it']
```

Fig.10 – Tokenizzazione in Python

Come possiamo notare dall'*output*, questo programma è stato in grado di riconoscere i casi in cui i simboli di punteggiatura dovevano essere presi singolarmente (":"), come parte di una parola ("U.S.A."), come parte di un prezzo ("€18,90") e come parte di un sito web ("www.cartoline.it").

La seconda fase della *text normalization* è la normalizzazione dei formati delle parole. Per normalizzazione delle parole si intende la trasformazione delle stesse in un formato standard, scegliendo un'unica forma per parole con forme multiple come USA e US, per permettere ai software di analizzarle al meglio, senza commettere errori. Nonostante si perdano alcune informazioni ortografiche, questa standardizzazione è molto preziosa, in quanto ci permette di prendere in

considerazione parole diverse che si riferiscono allo stesso concetto. Per esempio, se fossimo interessati ad estrarre informazioni sugli Stati Uniti, sarebbe estremamente utile se ci venissero mostrate tutte le informazioni contenute nei documenti a prescindere dal fatto che sia menzionato "US" o "USA".

Un altro tipo di normalizzazione è il *case folding* che consiste nel rappresentare nello stesso modo le lettere maiuscole e quelle minuscole. Questo strumento è molto utile in parecchi ambiti, come il reperimento di informazioni o il riconoscimento vocale. D'altro canto, per effettuare analisi come quella del *sentiment* o la traduzione automatica, le maiuscole e le minuscole possono essere molto utili e generalmente il *case folding* non viene eseguito. Questo perché il beneficio di mantenere la differenza tra, ad esempio, "US" inteso come paese e "*us*" inteso come pronome, è nettamente superiore rispetto al vantaggio nella generalizzazione che il *case folding* potrebbe fornire per altre parole.

Per molte situazioni di elaborazione del linguaggio naturale, può essere utile che due forme morfologicamente diverse di una parola si comportino in modo simile. Per esempio, nella ricerca sul web, qualcuno potrebbe digitare la stringa "serpente", ma gli sarebbe molto utile se la ricerca gli restituisse anche le pagine che menzionano "serpenti". Questo è particolarmente importante quando si effettuano ricerca in lingue morfologicamente complesse come il polacco, dove per esempio la parola *Warsaw* (Varsavia) ha suffissi diversi in base a se svolge il ruolo di soggetto (*Warszawa*), o se si trova dopo una preposizione come "*in Warsaw*" (*w Warszawie*), o "*to Warsaw*" (*do Warszawy*).

La *lemmatization* è proprio il compito di determinare se due parole hanno la stessa radice, nonostante le loro differenze superficiali. Per esempio, le parole "*am*", "*are*" e "*is*" hanno il lemma condiviso "*be*"; le parole "*dinner*" e "*dinners*" hanno entrambe il lemma "*dinner*". Utilizzando questa tecnica è possibile trovare, tornando all'esempio precedente, tutte le diverse parole polacche che fanno riferimento a "*Warsaw*". La forma lemmatizzata di una frase come "*He is reading detective stories*" sarebbe quindi "*He be read detective story*".

I metodi più sofisticati di lemmatizzazione prevedono il parsing[37] morfologico completo della parola. La morfologia è una disciplina linguistica che studia la struttura e la formazione delle parole. Si concentra sull'analisi dei morfemi, che sono le unità più piccole e significative delle parole, e come questi si combinano per formarne di nuove. Il morfema più importante è la radice, che fornisce il significato principale della parola. Seguono poi i prefissi (che si collocano prima della radice), i suffissi (che si aggiungono alla fine della radice) e gli infissi (che si inseriscono all'interno della radice) che aggiungono significati "aggiuntivi" di vario tipo.

[37] Algoritmo che, sulla base della grammatica e del lessico di una lingua data, effettua un'analisi automatica della struttura morfologica delle parole. *Dizionario Treccani.*

Gli algoritmi di lemmatizzazione possono essere però molto complessi. Per questo motivo, a volte si ricorre a un metodo più semplice ma più rozzo, che consiste principalmente nel tagliare i suffissi delle parole. Questa versione semplificata dell'analisi morfologica è chiamata *stemming*. L'algoritmo di *stemming* più famoso è il Porter *stemmer*[38], ideato dallo studioso M.F. Porter, da non confondere con l'economista Micheal Porter. Per rendere l'idea del funzionamento di tale algoritmo, riporto un esempio che ho effettuato con *Python*.

Scrivendo queste righe di codice:

```
import nltk
from nltk.stem import PorterStemmer
nltk.download("punkt")

ps=PorterStemmer()
words=['program','programming','programs', 'programmed']

for x in words:
    print(x+ '->'+ ps.stem(x))
```

Fig.11 – Porter stemmer in Python

L'output che otterremo sarà:

```
In [7]: runcell(1, '/Users/vincenzocamerlengo/Desktop
STEMMER.py')
program->program
programming->program
programs->program
programmed->program
```

Fig.12 – Output

La segmentazione delle frasi è l'ultima fase del processo di normalizzazione del testo. Come facilmente intuibile, il compito della *text segmentation* è quello di dividere i vari periodi di un testo e gli indizi più utili per svolgere questo lavoro sono ovviamente i simboli di punteggiatura, come i punti, i punti interrogativi e i punti esclamativi. I punti interrogativi ed esclamativi sono marcatori relativamente univoci dei confini della frase. I punti, invece, sono più ambigui. Tale carattere, infatti, può essere utilizzato anche per effettuare abbreviazioni come Mr. o ecc. Proprio questa ultima frase rappresenta un caso ancora più complesso di questa ambiguità, in cui il punto finale di "ecc." rappresenta sia un'abbreviazione che un marcatore di confine della frase. Per questo motivo, la tokenizzazione delle frasi e quella delle parole possono essere affrontate congiuntamente.

In generale, i metodi di tokenizzazione delle frasi hanno il compito di stabilire innanzitutto se un punto fa parte della parola o è un marcatore di confine della frase. A questo scopo, possedere un dizionario delle abbreviazioni più comunemente usate può essere estremamente utile. Tali dizionari possono essere costruiti a mano o appresi automaticamente. Per effettuare la segmentazione delle

[38] Porter, M. F. (1980). *An algorithm for suffix stripping. Program, 14(3), 130-137.*

frasi, il metodo più utilizzato è l'utilizzo dello *Stanford CoreNPL toolkit* secondo il quale una frase termina quando un simbolo di punteggiatura di fine frase (".", "!", "?") non fa parte di un *token* (come nel caso di un'abbreviazione o di un numero).

2.2.3 MINIMUM EDIT DISTANCE

La distanza di modifica minima (in inglese *Minimum Edit Distance*) è una metrica utilizzata per misurare la somiglianza tra due stringhe. Questa metrica conta il numero minimo di operazioni necessarie per trasformare una stringa in un'altra. Le operazioni che possono essere eseguite per trasformare una stringa in un'altra sono le seguenti: inserimento di un carattere, rimozione di un carattere, sostituzione di un carattere con un altro.

Questo metodo è molto importante ed è utilizzato principalmente da Microsoft Word per suggerire correzioni ortografiche e da Google per consigliare ricerche affini:

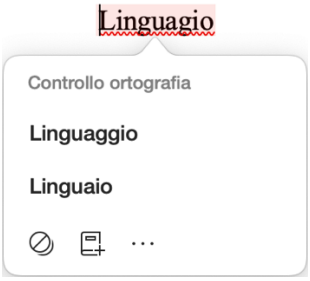


Fig.13 – Esempio in Word

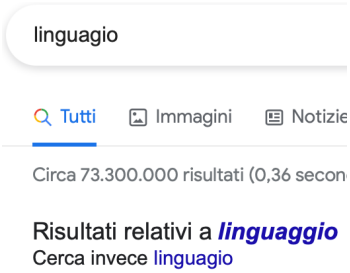


Fig.14 – Esempio in Google

Come spiegato in precedenza, tramite la *minimum edit distance* possiamo stabilire quanto una stringa sia simile ad un'altra. Per esempio, supponendo per semplificazione che ogni operazione abbia costo 1, la differenza tra la parola "*intention*" e la parola "*execution*" sarà di 5 operazioni dove "d" sta per "deletion", "s" per "substitution" e "i" per "insertion":

```
I N T E * N T I O N
| | | | | | | | | |
* E X E C U T I O N
d s s   i s
```

Fig.15 - Jurafsky D. & Martin J.H. (2023). Speech and Language Processing. Third edition.

Negli anni '60, lo scienziato russo Vladimir Levenshtein propose un altro metodo nel calcolo della *minimum edit distance* che non prevedeva l'operazione di sostituzione; di conseguenza, utilizzando

questo metodo, per sostituire un carattere con un altro è necessario effettuare due operazioni, prima quella di eliminazione e poi quella di inserimento. Pertanto, volendo calcolare la distanza tra le due parole dell'esempio precedente, il nuovo risultato sarà 8 (1+2+2+1+2).

Come si trova la distanza minima di modifica in casi molto più complessi di questo?

Lo spazio di tutte le possibili modifiche è enorme, quindi non possiamo fare una ricerca ingenua. Tuttavia, molti percorsi di modifica da una stringa all'altra potrebbero trovarsi, in uno stadio intermedio, ad essere identici; quindi, invece di ricordare tutti questi percorsi, potremmo semplicemente ricordare il percorso più breve verso uno stato ogni volta che lo vediamo. Per fare questo si può usare la programmazione dinamica.

La programmazione dinamica è una tecnica di progettazione di algoritmi che si basa sulla suddivisione di un problema in sotto-problemi più piccoli, risolvendoli in modo indipendente e memorizzando i risultati per evitare di doverli risolvere nuovamente in futuro.

Questa tecnica è molto utile per risolvere problemi di ottimizzazione, in cui si cerca di trovare la soluzione migliore tra molte possibili alternative. Il cuore della programmazione dinamica è la memoria degli stati precedenti, in modo da poter sfruttare i risultati già calcolati per risolvere problemi simili in modo più efficiente.

Si consideri il percorso più breve per trasformare la parola "*intention*" nella parola "*execution*" rappresentato nella figura sottostante.

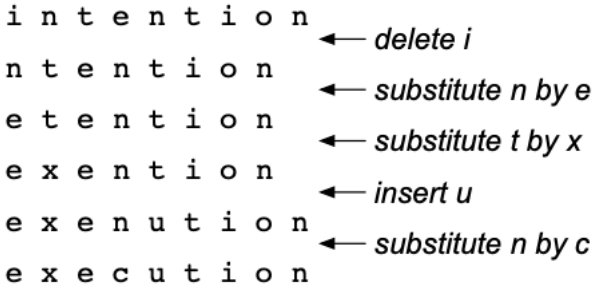


Fig.16 – Esempio

Prendiamo in esame una qualsiasi stringa che si trova in questo percorso ottimale, per esempio "*exention*". L'intuizione della programmazione dinamica è che se "*exention*" è nell'elenco delle operazioni ottimali, allora la sequenza ottimale per passare da "*intention*" a "*execution*" deve necessariamente includere anche il percorso ottimale che va da "*intention*" a "*exention*". Il motivo è facilmente intuibile; se infatti ci fosse un percorso più breve da "*intention*" a "*exention*", allora potremmo usarlo al posto di quello trovato in precedenza, ottenendo un percorso complessivo più breve, e la sequenza ottimale non sarebbe ottimale, portando così a una contraddizione.

L'algoritmo di *minimum edit distance* inizia creando una matrice di dimensioni $(m + 1) \times (n + 1)$, dove m e n sono le lunghezze delle due stringhe x e y da confrontare. Definiamo poi la cella $D(i, j)$

di questa matrice come la distanza di edit minima tra i primi i caratteri della stringa x e i primi j caratteri della stringa y.

L'algoritmo riempie questa matrice calcolando la distanza di modifica minima per ogni coppia di prefissi delle due stringhe. In particolare, per calcolare la distanza minima tra i primi i caratteri della stringa x e i primi j caratteri della stringa y, l'algoritmo prende in considerazione le seguenti tre operazioni:

- cancellazione: rimuovere l'ultimo carattere dalla prima stringa e confrontare il prefisso rimanente con i primi j caratteri della seconda stringa;
- inserimento: aggiungere un carattere alla fine della prima stringa e confrontarlo con i primi $j-1$ caratteri della seconda stringa;
- sostituzione: sostituire gli ultimi caratteri delle due stringhe.

L'algoritmo, quindi, sceglie la modifica che minimizza la distanza di *edit* tra i due prefissi e la applica alla cella corrispondente nella matrice secondo questa formula:

$$D(i,j) = min \begin{cases} D(i-1,j)+1 \\ D(i,j-1)+1 \\ D(i-1,j-1)+1 \end{cases} \tag{2.1}$$

Dove 1 rappresenta il costo di ogni operazione.

Nel caso in cui invece volessimo utilizzare la matrice di Levenshtein, dove l'operazione di sostituzione costa 2, la formula 2.1 diventa la seguente:

$$D(i,j) = min \begin{cases} D(i-1,j)+1 \\ D(i,j-1)+1 \\ D(i-1,j-1)+2 \end{cases} \tag{2.2}$$

Si completa così tutta la matrice che porta al calcolo finale della *minimum edit distance* che sarà rappresentata dalla cella $D(m+1, n+1)$. Si noti che la prima colonna e la prima riga rappresentano delle stringhe vuote; pertanto, la distanza di modifica tra quest'ultima ed i primi i caratteri della stringa sarà proprio i.

Src\Tar	#	e	x	e	c	u	t	i	o	n
#	0	1	2	3	4	5	6	7	8	9
i	1	2	3	4	5	6	7	6	7	8
n	2	3	4	5	6	7	8	7	8	7
t	3	4	5	6	7	8	7	8	9	8
e	4	3	4	5	6	7	8	9	10	9
n	5	4	5	6	7	8	9	10	11	10
t	6	5	6	7	8	9	8	9	10	11
i	7	6	7	8	9	10	9	8	9	10
o	8	7	8	9	10	11	10	9	8	9
n	9	8	9	10	11	12	11	10	9	8

Fig.17 - Jurafsky D. & Martin J.H. (2023). Speech and Language Processing. Third edition.

2.3 NAIVE BAYES TEXT CLASSIFICATION FOR SENTIMENT ANALYSIS

La classificazione è alla base dell'intelligenza umana e di quella delle macchine. Riconoscere volti o voci, smistare la posta, assegnare voti ai compiti: sono tutti esempi di assegnazione di una categoria a un *input*. Le potenziali sfide di questo compito sono evidenziate dallo scrittore argentino Jorge Luis Borges che immaginò di classificare gli animali in[39]:

"(a) quelli che appartengono all'imperatore, (b) quelli imbalsamati, (c) quelli addestrati, (d) i maialini da latte, (e) le sirene, (f) quelli favolosi, (g) i cani randagi, (h) quelli che sono inclusi in questa classificazione, (i) quelli che tremano come se fossero pazzi, (j) quelli innumerevoli, (k) quelli disegnati con un pennello di pelo di cammello molto fine, (l) altri, (m) quelli che hanno appena rotto un vaso di fiori, (n) quelli che assomigliano a mosche da lontano."

Molti compiti di elaborazione del linguaggio prevedono la classificazione, anche se fortunatamente le nostre classi sono molto più facili da definire rispetto a quelle di Borges. In questo paragrafo introduciamo l'algoritmo di Bayes (nella versione "*naive*" quindi semplice) e lo applichiamo alla categorizzazione del testo, il compito di assegnare un'etichetta o una categoria a un intero testo o documento.

In particolare, ci concentriamo sull'analisi del *sentiment*, ovvero l'orientamento positivo o negativo che un autore esprime nei confronti di un oggetto. Una recensione di un film, di un libro o di un prodotto sul web esprime il sentimento dell'autore verso il prodotto, mentre un testo editoriale o politico esprime il sentimento verso un candidato o un'azione politica. L'estrazione del *sentiment* da un testo è quindi importante in svariati campi che vanno dal marketing alla politica.

La versione più semplice della *sentiment analysis* è un compito di classificazione binaria e le parole delle recensioni forniscono ottimi spunti. Consideriamo, ad esempio, le seguenti frasi estratte da recensioni positive e negative di film e ristoranti. Aggettivi come "grande", "ricco", "patetico", "fantastico", "orribile" e "ridicolo" sono spunti molto informativi:

+ ...personaggi bizzarri e satira riccamente applicata, e alcuni grandi colpi di scena

- Era patetico. La parte peggiore sono state le scene di boxe...

+ ... una fantastica salsa al caramello e mandorle dolci e tostate. Adoro questo posto!

- ...pizza orribile e prezzi ridicoli...

Il rilevamento dello *spam* è un'altra importante applicazione che consiste nel compito di classificazione binaria di assegnare un'e-mail a una delle due classi "*spam*" o "non *spam*". Per eseguire questa classificazione si possono usare molte caratteristiche lessicali e di altro tipo. Ad esempio, si potrebbe ragionevolmente sospettare di un'e-mail contenente frasi come "In regalo per te", "SENZA ALCUN COSTO" o "Caro vincitore".

[39] Borges, J. L. (1937-1952). The analytical language of John Wilkins. *Other inquisitions*.

Un'altra cosa che potremmo voler sapere di un testo è la lingua in cui è scritto. I testi sui *social media*, ad esempio, possono essere scritti in un numero enorme di lingue e per ognuna di essa dovremo applicare un'elaborazione diversa. Il compito di identificare la lingua è quindi il primo passo nella maggior parte delle *pipeline* di elaborazione linguistica.

Naive Bayes è un algoritmo di apprendimento automatico utilizzato per classificare i testi in categorie predefinite. L'approccio di base di *naive Bayes* è quello di utilizzare la probabilità condizionata per stimare la probabilità che un dato testo appartenga a una determinata categoria.

Questo algoritmo si basa sull'assunzione "*naive*" (semplice) che le diverse caratteristiche del testo siano indipendenti tra loro, il che semplifica notevolmente i calcoli. In altre parole, assume che la presenza di una determinata parola nel testo non sia correlata alla presenza di altre parole.

Per utilizzare l'algoritmo di classificazione *naive Bayes* è necessario prima addestrare il modello utilizzando un set di dati di addestramento che contiene i testi da classificare e le rispettive categorie. Il modello utilizza quindi la probabilità condizionata per stimare la probabilità che un dato documento d appartenga a ciascuna delle categorie c possibili. Infine, il testo viene assegnato alla classe con la probabilità più alta che indicheremo con $\hat{c}$. Matematicamente è indicato dalla seguente formula:

$$\hat{c} = argmax\ P(c|d) \tag{2.3}$$

L'idea dell'inferenza Bayesiana è nota fin dai lavori di Bayes del 1763[40] ed è stata applicata per la prima volta alla classificazione dei testi da Mosteller e Wallace nel 1964[41]. L'intuizione della classificazione Bayesiana consiste nell'utilizzare la regola di Bayes, rappresentata dalla formula 2.4 per sviluppare l'equazione precedente.

$$P(x|y) = \frac{P(y|x)P(x)}{P(y)} \tag{2.4}$$

Sostituendo quindi l'equazione 2.4 nella 2.3 otterremo:

$$\hat{c} = argmax\ P(c|d) = argmax\ \frac{P(d|c)P(c)}{P(d)} \tag{2.5}$$

[40] Bayes, T. (1763). LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, FRS communicated by Mr. Price, in a letter to John Canton, AMFR S. *Philosophical transactions of the Royal Society of London*, (53), 370-418.

[41] Mosteller, F., & Wallace, D. L. (1963). Inference in an authorship problem: A comparative study of discrimination methods applied to the authorship of the disputed Federalist Papers. *Journal of the American Statistical Association*, *58*(302), 275-309.

Adesso è possibile semplificare l'equazione 2.5 eliminando il denominatore $P(d)$. Questo è possibile perché dovremo calcolare e confrontare $\frac{P(d|c)P(c)}{P(d)}$ per ogni c e ovviamente $P(d)$ non varia mai. Pertanto, l'equazione diventa:

$$\hat{c} = argmax\ P(d|c)P(c) \tag{2.6}$$

Ogni documento, inoltre, può anche essere rappresentato come un insieme di parole ("*words*") che indicheremo con $w_1,\ w_2, \dots ,\ w_n$. L'equazione 2.6 può quindi essere sviluppata come segue:

$$\hat{c} = argmax\ P(w_1,\ w_2, \dots ,\ w_n\ |c)P(c) \tag{2.7}$$

Purtroppo, l'equazione 2.7 è ancora troppo difficile da calcolare direttamente: senza alcune ipotesi semplificative, stimare la probabilità di ogni possibile combinazione di parole richiederebbe un numero enorme di parametri. I classificatori *naive Bayes* fanno quindi due ipotesi semplificative.

- La prima è l'ipotesi del *bag-of-words*: assumiamo che la posizione non abbia importanza e che ogni parola abbia lo stesso effetto sulla classificazione sia che si presenti come prima, ventesima o ultima parola del documento. Pertanto, assumiamo che le caratteristiche $w_1,\ w_2, \dots ,\ w_n$ codifichino solo l'identità della parola e non la posizione.
- La seconda ipotesi è detta "ipotesi di indipendenza condizionale" secondo la quale le probabilità $P(\ w_i|c)$ sono indipendenti data la classe c e quindi possono essere moltiplicate come segue:

$$P(w_1,\ w_2, \dots ,\ w_n\ |c) = P(w_1|c) \cdot P(w_2|c) \cdot \dots \cdot P(w_n|c) \tag{2.8}$$

Per applicare il classificatore di Bayes al testo, dobbiamo considerare le posizioni delle parole e ciò è possibile scorrendo un indice i attraverso ogni posizione delle parole nel documento. Sostituendo infine la 2.8 nella 2.7, l'equazione finale diventa:

$$\hat{c} = argmax\ P(c) \prod_{i\ \epsilon\ positions} P(w_i|c) \tag{2.9}$$

Adesso una domanda sorge spontanea: come si calcolano realmente queste probabilità?
Per il calcolo di $P(c)$ bisogna trovare la percentuale di documenti presenti nel nostro set di addestramento che appartiene a ciascuna classe c. Definendo con N_c il numero di documenti con classe c e con N_{doc} il numero totale di documenti, avremo:

$$\hat{P}(c) = \frac{N_c}{N_{doc}} \tag{2.10}$$

Per calcolare invece la probabilità $P(w_i|c)$, bisogna trovare il numero di volte in cui la parola w_i appare tra tutte le parole dei documenti di argomento c. Per prima cosa concateniamo tutti i documenti con categoria c in un unico grande testo "categoria C" e definiamo invece con V l'unione di tutte le parole presenti in tutte le classi. La probabilità che una parola w_i faccia parte di una classe c sarà data dal rapporto tra il numero di volte in cui si trova in C ed il numero di volte in cui si trova in V:

$$\hat{P}(w_i|c) = \frac{count(w_i, C)}{count(w, V)} \tag{2.11}$$

Sebbene la classificazione testuale *naive Bayes* possa funzionare bene per l'analisi del *sentiment*, in genere si utilizzano alcune piccole modifiche per migliorarne le prestazioni.
In primo luogo, per la classificazione del *sentiment* e per una serie di altri compiti di classificazione del testo, la presenza o meno di una parola sembra essere più importante della sua frequenza. Per questo motivo, spesso le prestazioni migliorano se il conteggio delle parole in ogni documento è pari a 1. Si procede quindi ad eliminare tutte le parole duplicate all'interno di ogni documento; questa variante è chiamata *binary multinomial naive Bayes* o *binary naive Bayes*. La figura seguente mostra un esempio in cui un insieme di quattro documenti (abbreviati e normalizzati) viene "rimappato" in binario, con i conteggi modificati mostrati nella tabella a destra. Si noti che i conteggi dei risultati non devono necessariamente essere pari a 1 in quanto vengono eliminati i duplicati soltanto all'interno di ogni documento. Pertanto, se una parola è contenuta in più documenti il suo conteggio sarà superiore ad uno.

Four original documents:

− it was pathetic the worst part was the boxing scenes
− no plot twists or great scenes
+ and satire and great plot twists
+ great scenes great film

After per-document binarization:

− it was pathetic the worst part boxing scenes
− no plot twists or great scenes
+ and satire great plot twists
+ great scenes film

	NB Counts		Binary Counts	
	+	−	+	−
and	2	0	1	0
boxing	0	1	0	1
film	1	0	1	0
great	3	1	2	1
it	0	1	0	1
no	0	1	0	1
or	0	1	0	1
part	0	1	0	1
pathetic	0	1	0	1
plot	1	1	1	1
satire	1	0	1	0
scenes	1	2	1	2
the	0	2	0	1
twists	1	1	1	1
was	0	2	0	1
worst	0	1	0	1

Fig.18 - Jurafsky D. & Martin J.H. (2023). Speech and Language Processing. Third edition.

Una seconda importante aggiunta che viene comunemente fatta quando si effettua la classificazione del testo per la *sentiment analysis* è quella di trattare la negazione. Consideriamo la differenza tra "mi piace molto questo film" (positivo) e "non mi è piaciuto questo film" (negativo). La negazione espressa da "non" altera completamente il significato che traiamo dal predicato "piacere". Allo stesso modo, la negazione può modificare una parola negativa per produrre una recensione positiva ("non scartare questo film, non ci fa annoiare").

Una linea di base molto semplice, comunemente utilizzata nella *sentiment analysis* per gestire la negazione, è la seguente: durante la normalizzazione del testo, si antepone il prefisso "*NOT_*" a ogni parola che si trova dopo un token di negazione logica fino al successivo segno di interpunzione. Così la frase:

"*Didn't like this movie, but I…*" diventa "*Didn't NOT_like NOT_this NOT_movie, but I…*"

Le "parole" di nuova formazione come "*NOT_like*" si presenteranno quindi più spesso in documenti negativi e agiranno come spunti per il sentimento negativo, mentre parole come "*NOT_bored*", "*NOT_dismiss*" acquisiranno associazioni positive.

Infine, in alcune situazioni potremmo non avere dati sufficienti per addestrare i classificatori *naive Bayes*. In questi casi, possiamo ricavare le caratteristiche delle parole positive e negative da lessici del sentimento, cioè elenchi di parole pre-annotate con sentimento positivo o negativo. I tre lessici del sentimento più famosi sono il General Inquirer[42], il LIWC[43]e il lessico della soggettività MPQA[44]. Ad esempio, quest'ultimo contiene 6885 parole, ognuna delle quali è contrassegnata da segno "+" o "-" per indicare se ha polarità positiva o negativa.

Poste ormai tutte le basi teoriche che si trovano dietro all'analisi ed alla manipolazione dei testi da parte dei *software* di *text mining*, procederemo nel prossimo capitolo effettuando la ricerca del *sentiment* su tre diverse imprese quotate in borsa ed in infine, nell'ultimo capitolo, verificheremo l'esistenza di una correlazione tra il *sentiment* degli utenti verso queste imprese ed i rendimenti dei loro titoli.

[42] Stone, P. J., Dunphy, D. C., & Smith, M. S. (1966). The general inquirer: A computer approach to content analysis.
[43] Pennebaker, J. W., Francis, M. E., & Booth, R. J. (2001). Linguistic inquiry and word count: LIWC 2001. *Mahway: Lawrence Erlbaum Associates*, *71*(2001), 2001.
[44] Wilson, T., Wiebe, J., & Hoffmann, P. (2005, October). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of human language technology conference and conference on empirical methods in natural language processing* (pp. 347-354).

CAPITOLO TERZO

RICERCA DEL SENTIMENT

3.1 IL RUOLO DEI SOCIAL MEDIA

Al giorno d'oggi i *social media* rivestono un ruolo particolarmente importante nella quotidianità di ognuno di noi. Per fare un confronto con la vita senza *social networks* non serve andare indietro di troppi anni e, da una quindicina di anni a questa parte, siamo tempestati continuamente da notizie e notifiche che ci giungono sui nostri *smartphone* e sui nostri computer. Internet è infatti diventato il mezzo preferito, oltre che più accessibile ed accurato, per informarsi istantaneamente su quel che accade in ogni parte del mondo.

Nei grafici[45] riportati in basso è possibile osservare l'evoluzione dell'utenza complessiva di ogni strumento di informazione dal 2007 al 2021:

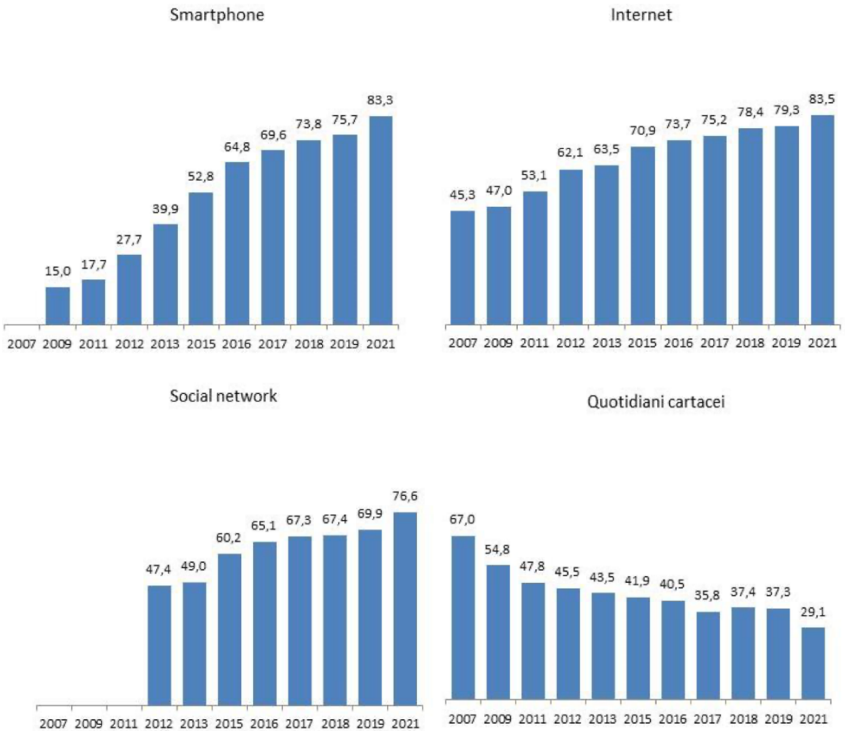
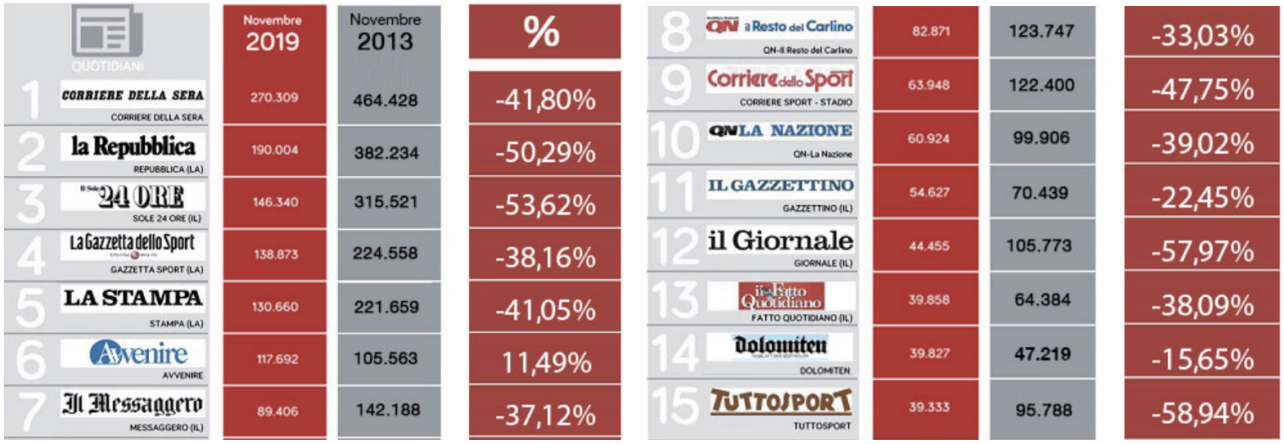


Fig.19 - Censis. 17º Rapporto sulla comunicazione, "I media dopo la pandemia". Roma, 6 ottobre 2021

Al pari con l'evoluzione digitale, stiamo assistendo anche ad una constante diminuzione dell'uso dei giornali cartacei. La motivazione è molto semplice; potendo consultare internet in ogni istante della nostra giornata, le notizie che usciranno sui giornali il giorno dopo saranno ritenute già vecchie e conosciute.

[45] Censis. 17º Rapporto sulla comunicazione, "I media dopo la pandemia". Roma, 6 ottobre 2021

Nel 2022, nel corso del tradizionale raduno tra ex alunni della *Columbia University*, è intervenuto sul tema il direttore del Los Angeles Times, Kevin Merida, affermando:
"Io amo la carta e mi piace averla tra le mani, sfogliarla, scegliere la sezione da leggere. Mi piace la cura con cui la stampiamo. Però è inutile prenderci in giro: i giornali di carta spariranno tra uno e massimo cinque anni di tempo. La grande maggioranza diventerà solo digitale o chiuderà".
Anche il quotidiano online "Il Post" ha approfondito questo tema pubblicando il confronto tra il numero di giornali venduti nel 2019 e quelli venduti nel 2013. I dati più aggiornati sono dell'anno 2020 ma, essendo stato questo anno fortemente influenzato dall'avvento della pandemia da COVID-19, analizzeremo appunto i dati del 2019[46] che riescono comunque a darci una prova di quanto la vendita delle testate giornalistiche stia andando a picco.



QUOTIDIANI	Novembre 2019	Novembre 2013	%
1 CORRIERE DELLA SERA	270.309	464.428	-41,80%
2 la Repubblica	190.004	382.234	-50,29%
3 24 ORE	146.340	315.521	-53,62%
4 La Gazzetta dello Sport	138.873	224.558	-38,16%
5 LA STAMPA	130.660	221.659	-41,05%
6 Avvenire	117.692	105.563	11,49%
7 Il Messaggero	89.406	142.188	-37,12%
8 il Resto del Carlino	82.871	123.747	-33,03%
9 Corriere dello Sport	63.948	122.400	-47,75%
10 LA NAZIONE	60.924	99.906	-39,02%
11 IL GAZZETTINO	54.627	70.439	-22,45%
12 il Giornale	44.455	105.773	-57,97%
13 il Fatto Quotidiano	39.858	64.384	-38,09%
14 Dolomiten	39.827	47.219	-15,65%
15 TUTTOSPORT	39.333	95.788	-58,94%

Fig.20 - https://www.ilpost.it/2020/01/15/diffusione-copie-quotidiani-2019/

Dalla seguente tabella emerge il terrificante dato che rispetto al 2013, nel 2019 le vendite di ogni testata sono praticamente dimezzate. Queste informazioni, pertanto, rafforzano ancora di più la dichiarazione di Kevin Merida, conferendogli un taglio più tangibile della drammaticità del calo delle vendite dei giornali cartacei.

Poste queste premesse, è chiaro che i *social media* ed internet in generale stanno "rubando" ai giornali il compito di divulgare le informazioni e proprio per questo motivo stanno diventando molto importanti ed influenti nella vita di ogni persona.
La comparsa di internet nell'esistenza delle persone rappresenta uno dei fenomeni che ha avuto un maggiore impatto sulla socialità degli ultimi vent'anni. In particolare, l'introduzione dei *social network* ha creato le condizioni ideali per una vera e propria rivoluzione all'interno della rivoluzione. I cambiamenti introdotti da queste nuove piattaforme digitali sono evidenti e radicali poiché

[46] https://www.ilpost.it/2020/01/15/diffusione-copie-quotidiani-2019/

coinvolgono completamente la vita sociale degli utenti, il loro modo di interagire con gli altri, il loro modo di pensare, di agire e persino di educarsi.

Per avere un'idea ancora più precisa, si pensi che il tempo medio globale trascorso sulle piattaforme social è di 142 minuti al giorno[47]. Questo valore è generato da una vasta gamma di utenti e utilizzi diversi, ma allo stesso tempo simili tra loro. I "Millennials", quelli nati cioè tra l'inizio degli anni '80 e la metà degli anni '90 e noti anche come "Generazione Y", costituiscono certamente uno dei bacini d'utenza più ampi e con maggiori competenze. Tuttavia, la mia generazione, la "Generazione Z", nata con o addirittura dopo l'introduzione di alcuni *social network*, merita una menzione particolare. Noi giovani siamo immersi in questa nuova modalità di vivere i *social*: interagiamo, comunichiamo e contribuiamo a creare nuove "realtà", come gli sport virtuali e le piattaforme di streaming.

I "Millennials" e la "Generazione Z" utilizzano i *social network* non solo come strumento di comunicazione, ma sempre di più come strumento di trasmissione e collaborazione tra reti di persone, comunità e organizzazioni potenziate da funzionalità tecnologiche.

Ciò che differenzia la comunicazione nei *social media* da quella dei mezzi di comunicazione tradizionali, come la televisione e la radio, è il fatto che i *social media* permettono una comunicazione interattiva. Offrono la possibilità di partecipare alla conversazione e non essere semplici ascoltatori passivi. Le persone non si limitano ad entrare nelle comunità online, ma vi partecipano attivamente apportando il loro contributo. Ad esempio, come è possibile notare nel grafico sottostante, negli ultimi anni gli utenti di YouTube caricano circa 500 ore di video al minuto[48].

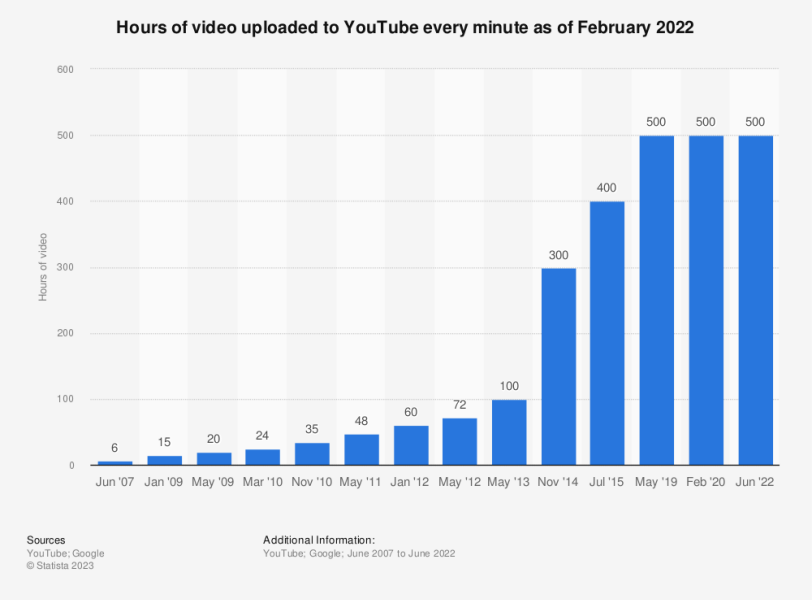


Fig.21 – Ore di video caricate ogni minuto su YouTube – fonte Statista

[47] https://www.statista.com/chart/26272/global-average-daily-time-spent-on-social-media-per-internet-user/
[48] https://www.statista.com/statistics/259477/hours-of-video-uploaded-to-youtube-every-minute/

Nel corso degli ultimi due anni, pesantemente segnati dalla pandemia, abbiamo assistito all'ascesa e alla caduta di diverse piattaforme social. Nonostante su Facebook ogni minuto vengano pubblicati 510.000 commenti, 293.000 post e 136.000 foto[49], le generazioni più giovani hanno rivolto lo sguardo verso nuove piattaforme.

Instagram è ormai una delle piattaforme *social* più famose e redditizie, utilizzata soprattutto da chi ha meno di 35 anni[50]. Altra piattaforma molto popolare è TikTok che, con i suoi 655,9 milioni di utenti nel 2021 con una crescita stimata fino a 955,3 milioni nel 2025[51], ha conquistato il palcoscenico con i suoi video di breve durata.

L'unica piattaforma *social* "longeva", che non sembra avere alcuna flessione, è Twitter. Si pensi che su questo *social media* vengono pubblicati circa 500 milioni di tweet al giorno, il che equivale a 6.000 tweet al secondo (Mention, 2018).

I *social media* ed internet in generale influenzano, e non di poco, i comportamenti, le idee e le decisioni di acquisto di ognuno di noi. Secondo un rapporto di Deloitte[52], il 29% degli utenti *social* ha maggiori probabilità di effettuare un acquisto lo stesso giorno di utilizzo dei *social media*; ciò significa che quando un utente vede un prodotto sui *social* è spinto ad acquistarlo cliccando semplicemente su un link o recandosi in un negozio fisico.

Questi sono solo alcuni dati che dimostrano l'ampio utilizzo dei nuovi strumenti di comunicazione e l'importante influenza che hanno nei nostri processi decisionali.

Proprio in virtù di questo ultimo punto, è chiaro che in internet è possibile trovare una miriade di informazioni che possono essere indicatori precoci e previsionali di un qualsiasi studio. Nel nostro caso, come chiaro dal titolo del lavoro, verificheremo l'esistenza di una correlazione tra il *sentiment*, ottenuto dalle notizie presenti sul web, ed il rendimento percentuale giornaliero dei titoli di un'impresa quotata in borsa. Per ottenere un risultato più preciso, oltre al *sentiment* utilizzeremo come regressore anche i volumi di ricerca su Google. Le imprese che prenderemo in esame sono tre: Amazon (AMZN), Tesla (TSLA) ed Apple (AAPL). Il mercato di riferimento è il NASDAQ, espresso in dollari statunitensi e l'arco temporale stabilito è compreso tra il 01/02/2022 ed il 22/04/2023.

Prima di procedere alla parte pratica del lavoro, nei successivi due paragrafi introdurrò la piattaforma "Refinitiv Eikon Workspace", grazie alla quale ho estrapolato il sentiment, ed il sito "Google Trends", dove ho ottenuto i dati relativi ai volumi di ricerca.

[49] https://blog.microfocus.com/how-much-data-is-created-on-the-internet-each-day/
[50] https://www.statista.com/statistics/248769/age-distribution-of-worldwide-instagram-users/
[51] https://www.statista.com/statistics/1327116/number-of-global-tiktok-users/
[52] https://www2.deloitte.com/content/dam/Deloitte/us/Documents/consumer-business/us-cb-navigating-the-new-digital-divide-051315.pdf

3.2 REFINITIV EIKON WORKSPACE & MARKETPSYCH

Refinitiv è un'azienda detenuta dal LSGE (London Stock Exchange Group) ed è uno dei maggiori fornitori mondiali di dati e infrastrutture dei mercati finanziari con 6,25 miliardi di dollari di entrate, oltre 40.000 clienti e 400.000 utenti finali in 190 paesi[53].

Nello specifico, Refinitiv Workspace è una piattaforma di dati e analisi finanziarie che fornisce una vasta gamma di strumenti per la gestione dei dati di mercato, la ricerca, l'analisi e la collaborazione in tempo reale, oltre ad offrire molteplici funzionalità per la visualizzazione e l'analisi dei dati di mercato, tra cui grafici interattivi, news e analisi di mercato, strumenti di analisi tecnica e fondamentale e flussi di dati in tempo reale.

La sezione più interessante di questa piattaforma è quella relativa al "News monitor"; questo strumento consente agli utenti di monitorare le notizie riguardanti un determinato titolo in tempo reale da una vasta gamma di fonti, tra cui agenzie di stampa, siti web di notizie e social media. Agli utenti, inoltre, è data la possibilità di configurare filtri personalizzati per selezionare le fonti di notizie e le categorie di interesse, come l'industria, la regione geografica o il tipo di evento, e ricevere notifiche in tempo reale quando vengono pubblicate nuove notizie rilevanti.

Inoltre, Refinitiv Workspace utilizza anche tecniche di analisi del *sentiment* per rilevare il tono e l'emozione delle notizie e degli articoli. Ciò consente agli utenti di identificare rapidamente le tendenze di mercato e le opinioni degli investitori su determinati strumenti finanziari o eventi di mercato. Ad esempio, gli utenti possono utilizzare gli strumenti di analisi del *sentiment* per identificare i titoli che suscitano maggiori reazioni positive o negative tra gli investitori e agire di conseguenza. Per effettuare la *sentiment analysis*, Refinitiv si affida ad un'altra piattaforma chiamata "MarketPsych".

MarketPsych è una piattaforma di analisi del *sentiment* e di dati di mercato basata sull'analisi del linguaggio naturale (NLP). La piattaforma è stata sviluppata da MarketPsych LLC, una società di ricerca e consulenza finanziaria specializzata nella misurazione dei fattori psicologici che influenzano i mercati finanziari. MarketPsych utilizza quindi algoritmi di NLP per analizzare i testi e valutarne il tono e l'emozione. In particolare, il sistema di analisi del *sentiment* utilizza algoritmi di *machine learning* per identificare le parole chiave e le espressioni che indicano una valutazione positiva o negativa di una notizia o di un'azienda. Proprio come abbiamo visto nel capitolo precedente dove abbiamo approfondito il *text mining*, questo sistema utilizza una vasta gamma di tecniche di elaborazione del linguaggio naturale, tra cui l'analisi semantica, la classificazione del testo e l'elaborazione delle entità, per identificare le parole e le frasi che indicano sentimenti positivi o negativi. Ad esempio, il sistema può identificare parole come "buono", "forte" ed "eccellente" come

[53] https://www.refinitiv.com/en/about-us

indicatori di un *sentiment* positivo, mentre parole come “cattivo”, “debole” e “pessimo” come indicatori di un *sentiment* negativo.

3.3 GOOGLE TRENDS

Google Trends è un programma che permette di vedere, praticamente in tempo reale, gli argomenti ed i termini di ricerca più popolari tra gli utenti Google. I ricercatori possono usare queste informazioni per esplorare potenziali idee per articoli e servizi oltre che per approfondire i dati tendenziali per mostrare il livello d’interesse generale, ad esempio, su un candidato politico, una tematica o un evento.

L’*homepage* di Google Trends riporta gli argomenti che Google rileva essere correlati e di tendenza nelle ricerche globali, in Google News o su YouTube. Le tendenze sono raccolte dalla tecnologia *Knowledge Graph* di Google, che unisce le informazioni sulle ricerche Google per rilevare gli argomenti più popolari in base ai picchi ed al volume assoluto delle ricerche.

Oltre ad osservare i temi di tendenza, è possibile effettuare delle ricerche specifiche riguardo un determinato argomento o parola chiave per analizzarne i volumi di ricerca nel corso del tempo. Inoltre, è possibile aggiungere svariati filtri come l’area geografica di interesse, l’arco temporale stabilito e la categoria di appartenenza del termine ricercato. Per esempio, supponiamo di voler essere interessati ad analizzare i volumi di ricerca su Google del termine “elezioni” in Italia nel corso degli ultimi 12 mesi. L’output che otterremo sarà il seguente:

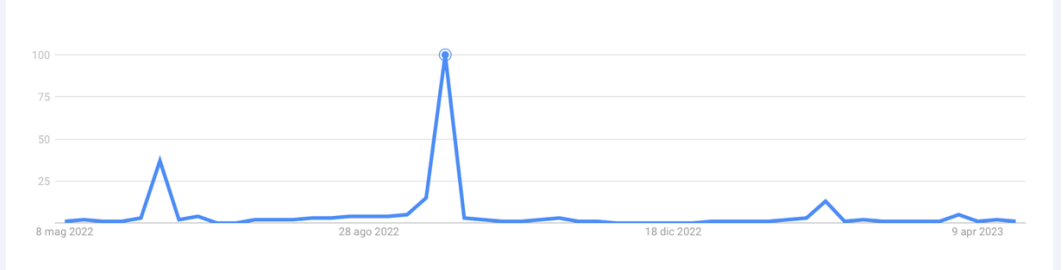


Fig.22 – Serie storica del termine “Elezioni” su Google Trends

Come facilmente pronosticabile, il picco delle ricerche si è registrato nella settimana tra il 25/09/2022 ed il 01/10/2022 proprio in vista delle elezioni politiche che si sono tenute in Italia il 25 settembre 2022.

È molto importante specificare che i risultati di Google Trends non sono riportati in termini assoluti ma sono normalizzati rispetto al valore più alto registrato, al quale verrà assegnato il valore 100.

Tra le altre funzionalità, Google Trends permette anche di confrontare fino a cinque termini alla volta in un unico grafico. Per esempio, possiamo ricercare i volumi di ricerca degli ultimi 5 anni degli sport più popolari in Italia (tennis, calcio, nuoto, basket e pallavolo; rappresentati rispettivamente da una linea blu, rossa, gialla, verde e viola), ottenendo questo risultato:

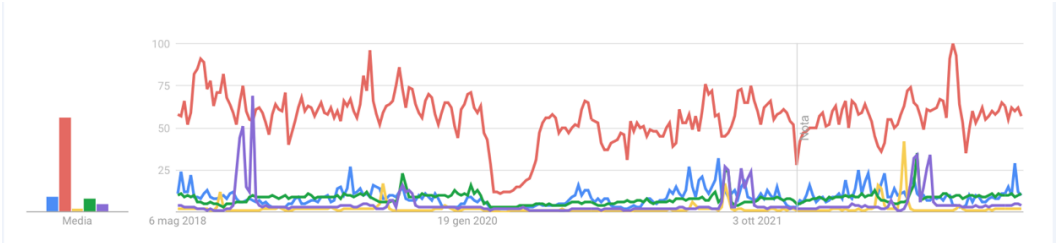


Fig.23 – Serie storiche su Google Trends

Focalizzandoci sull'obiettivo del presente lavoro, è chiaro che i dati estratti da Google Trends riguardo i volumi di ricerca di un determinato titolo azionario, potrebbero essere un ottimo parametro per stimare i rendimenti dei titoli stessi. Tutto ciò per la logica secondo il quale se molti utenti ricercano un titolo su Google, è molto probabile che siano interessati a scambiarlo sul mercato.

Nel prossimo paragrafo illustrerò le modalità grazie alle quali ho estratto ed esportato i dati da Refinitiv e da Google Trends a Excel, per poi ordinarli e manipolarli per renderli idonei alla generazione di serie temporali.

3.4 GENERAZIONE DI SERIE TEMPORALI

Nel seguente paragrafo illustrerò nello specifico gli step seguiti per ottenere tutti i dati utili alla ricerca riguardo l'azienda "Amazon" (è sottinteso che tutte le operazioni che verranno mostrate sono state eseguite anche riguardo le altre due aziende in esame).

3.4.1 SENTIMENT

Il primo passo è stata l'estrazione dei dati del *sentiment* dalla piattaforma Refinitiv. Come spiegato in precedenza, per effettuare questa operazione ci serviamo del "*News monitor*" della suddetta piattaforma. Nella barra di ricerca ho selezionato il titolo in questione, AMZN.O, anziché inserire una parola chiave come "Amazon" che avrebbe riportato nei risultati anche notizie inerenti all'azienda ma non al suo titolo nello specifico. Tra i principali filtri di ricerca che è possibile utilizzare troviamo: la lingua delle *news*, le fonti ed il periodo di riferimento. Nel mio caso ho preferito effettuare una ricerca esclusivamente in lingua inglese, sia perché è la lingua più diffusa al mondo, sia perché Refinitiv non è in grado di rilevare il *sentiment* da alcune lingue particolarmente complesse. Ho poi

selezionato tutte le fonti disponibili, salvo poi filtrarle per le più importanti onde evitare di prendere in analisi notizie di dubbia attendibilità. Il periodo di riferimento, come anticipato in precedenza, è 01/02/2022 – 22/04/2023.

Una volta ottenuti i dati li ho copiati ed esportati in un foglio di lavoro Excel per poterci lavorare.
Nella colonna del *sentiment* viene riportato se la rilevazione è "Mostly Positive", "Mostly Negative" oppure "Balanced". Per convenzione, ad ognuna di queste diciture ho assegnato rispettivamente i valori 1, -1 e 0. Pertanto, i valori del *sentiment* oscilleranno tra -1 ed 1 che rappresentano quindi le due polarità estreme, mentre 0 rappresenta un *sentiment* "neutrale".
Per effettuare questa operazione ho scritto il seguente comando nella cella E4 per poi trascinarla fino alla fine del foglio:

```
=+SE(D4="Mostly Negative";-1;SE(D4="Mostly Positive";1;SE(D4="Balanced";0;"")))
```

Fig.24 – Comando in Excel

In parole semplici, con questo comando ho chiesto ad Excel di verificare cosa contenesse la cella D4 e stampare in E4 il valore corrispondente.

Arrivati a questo punto sono passato al calcolo della media giornaliera dei rilevamenti del *sentiment*. Questa operazione, apparentemente semplice, risulta molto difficile quando si ha a che fare con migliaia di dati. Per effettuarla ho creato una macro su Excel, che consiste nella scrittura di un codice nel linguaggio di programmazione "Virtual Basic for Applications" o VBA, al fine di istruire il foglio ad eseguire una determinata azione. Il programma che ho implementato a questo fine è il seguente:

```
Sub media()

Dim sum, num As Integer


Range("A1").Select

For Count = 1 To 5373

sum = 0
num = 0

   If (IsDate(ActiveCell.Value)) Then
      sum = ActiveCell.Offset(2, 4).Value
      ActiveCell.Offset(1, 0).Select
      Do Until Not (IsEmpty(ActiveCell))
         sum = sum + ActiveCell.Offset(2, 4).Value
         num = num + 1
         ActiveCell.Offset(1, 0).Select
      Loop

      ActiveCell.Offset((-2 * num) - 1, 6) = sum / num

   End If
Next

End Sub
```

Fig.25 – Macro su Excel

Una volta applicata questa macro di nome “media” al nostro foglio di lavoro, ed aver utilizzato la funzione CONTA.NUMERI nella colonna E per visualizzare quante rilevazioni del *sentiment* abbiamo estratto nel nostro arco temporale, il foglio risulta completo per continuare l’analisi.

Infine, ho copiato su un altro foglio di lavoro le colonne riportanti la data e la rilevazione media del *sentiment*, in modo tale da ottenere una tabella riassuntiva facilmente consultabile.

In conclusione, è stato possibile creare la serie temporale dell’andamento del *sentiment* dal 01/02/2022 al 22/04/2023.

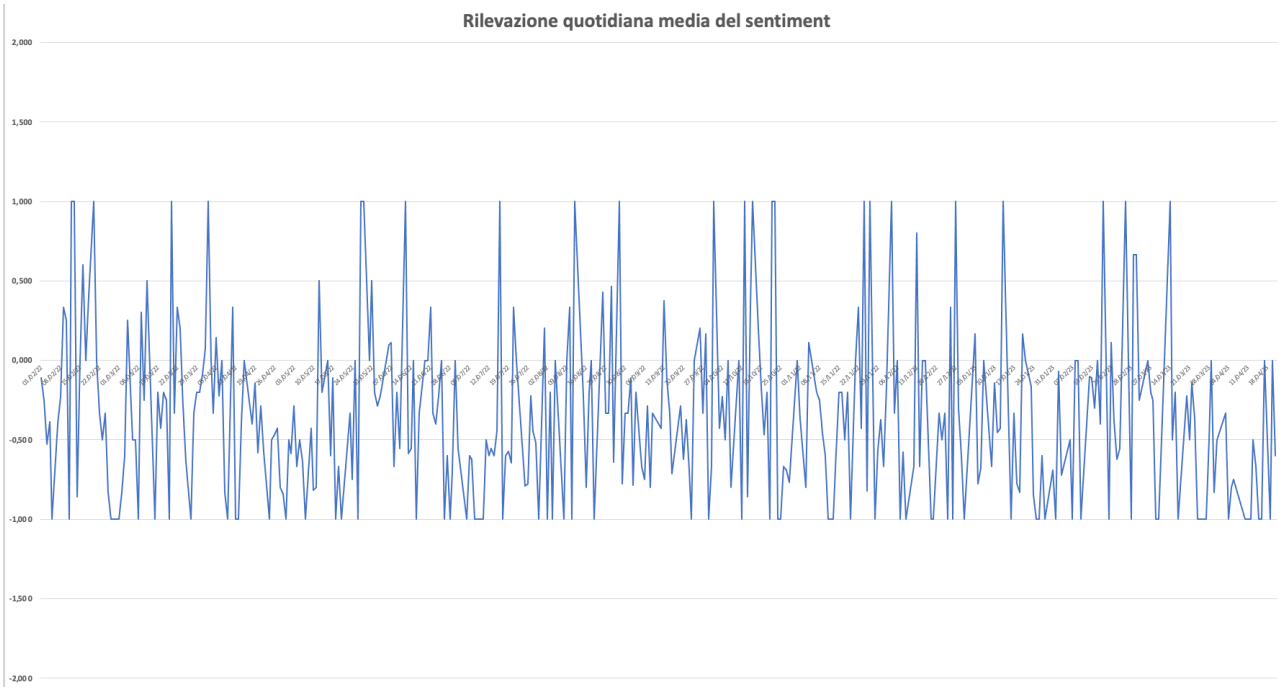


Fig.26 – Rilevazione quotidiana del sentiment

3.4.2 VOLUMI DI RICERCA GOOGLE

Estrapolare i dati da Google Trends è invece molto più semplice e intuitivo.

Come mostrato in precedenza basta cercare un termine, nel nostro caso AMZN, ed inserire il periodo di riferimento al quale siamo interessati. Ci apparirà un grafico con un pulsante in alto a destra che ci permette di esportare i dati in formato “.csv” per poterli analizzare.

A questo punto ho semplicemente copiato i dati in un foglio di lavoro Excel per poter creare la seguente serie temporale dei volumi di ricerca su Google:

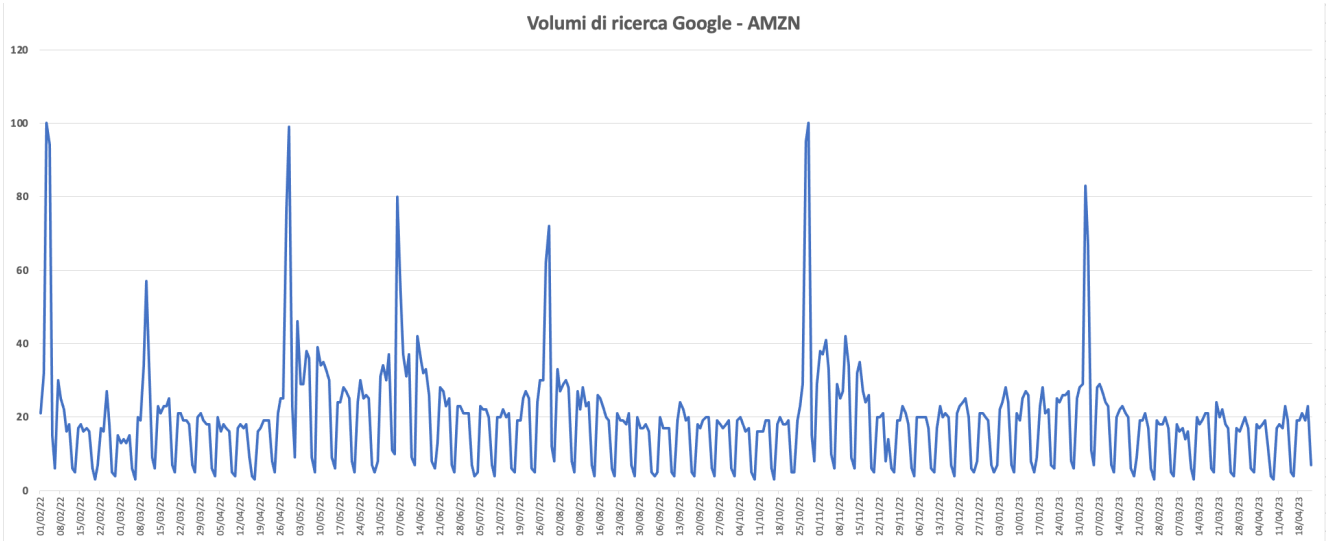


Fig.27 – Volumi di ricerca su Google del termine Amazon

3.4.3 RENDIMENTI GIORNALIERI

I rendimenti giornalieri di un determinato titolo sono facilmente accessibili da innumerevoli siti specializzati. Per effettuare però la ricerca più precisa possibile, ho deciso di affidarmi nuovamente alla piattaforma Refinitiv.

Nella sezione dedicata all'analisi tecnica ho effettuato la ricerca del titolo AMZN.O ottenendo il grafico dell'andamento del prezzo nell'intervallo temporale stabilito. Come è possibile notare dal grafico sottostante, ho inserito un'ulteriore analisi che rappresenta il rendimento giornaliero del titolo espresso in percentuale, che è ciò al quale siamo interessati.

Fig.28 - Refinitiv

Premendo poi sul pulsante cerchiato in rosso, è possibile esportare i dati direttamente su Excel per poter poi creare, come abbiamo visto in precedenza, la serie temporale.

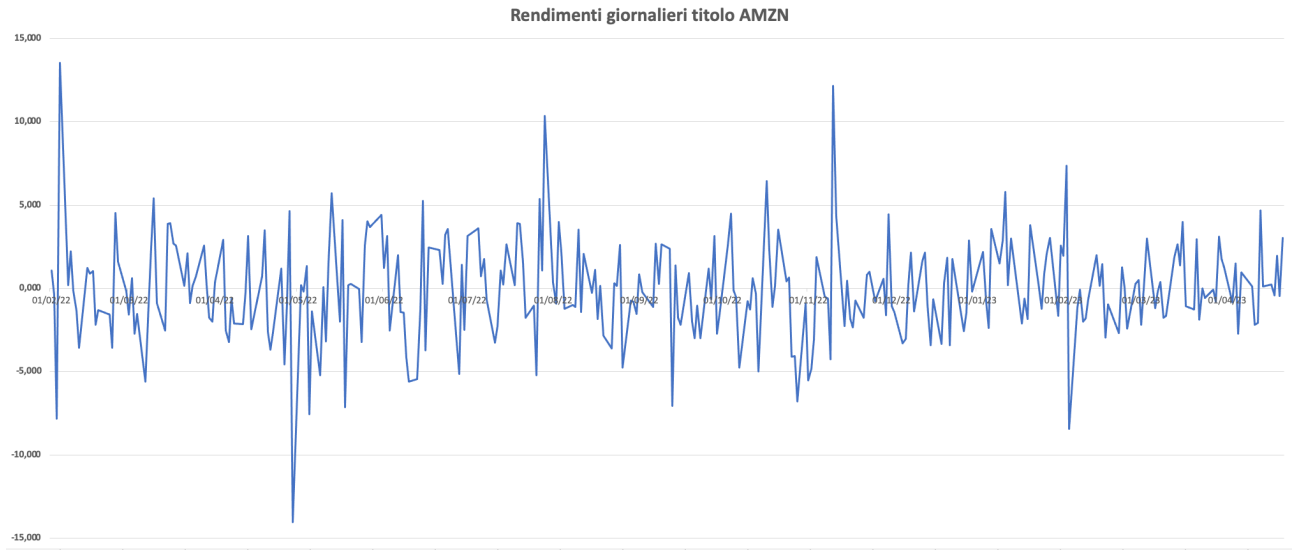


Fig.29 – Rendimenti giornalieri del titolo AMZN

Nel prossimo capitolo ci serviremo della piattaforma R per effettuare test statistici utili alla verifica del modello che stiamo costruendo.

CAPITOLO QUARTO

ESISTE UNA CORRELAZIONE TRA IL SENTIMENT ED I RENDIMENTI DEI TITOLI?

4.1 SERIE STORICHE

Prima di procedere all'analisi ed alla verifica di una possibile correlazione tra il *sentiment* ed i rendimenti dei titoli che abbiamo preso in esame, risulta doveroso introdurre teoricamente le serie storiche (o temporali) in modo da poterne comprendere più agevolmente i vari aspetti.

Prima di tutto, in statistica descrittiva, con il termine "serie" si intende la classificazione di diverse osservazioni di un fenomeno rispetto ad un carattere qualitativo; se tale carattere è il tempo, la serie viene detta *storica* o *temporale*.
Questa particolare tipologia di serie può essere estremamente utile per stimare l'effetto causale su una variabile di interesse Y, di un cambiamento in un'altra variabile X nel corso del tempo.

Il valore osservato dalla variabile Y al tempo t è indicato con Y_t ed il numero totale delle osservazioni è indicato con T. La differenza del valore di Y tra il periodo $t-1$ ed il periodo t è indicato da $Y_t - Y_{t-1}$ che rappresenta la cosiddetta "differenza prima".
Il valore di Y nel periodo precedente, rispetto al periodo corrente t, è chiamato "primo valore ritardato", o più comunemente "primo ritardo" e si indica con Y_{t-1}. In generale, il j-esimo ritardo di Y, è rappresentato dal suo valore j periodi indietro nel tempo e si indica con Y_{t-j}. [54]

Altro concetto molto importante è quello dell'autocorrelazione. Con questo termine si indica la correlazione di una serie con i propri valori ritardati. La prima autocorrelazione è la correlazione tra cioè, la correlazione tra valori di Y in tempi adiacenti. La seconda autocorrelazione è la correlazione tra Y_t e Y_{t-2}, e la j-esima autocorrelazione è la correlazione tra Y_t e Y_{t-j}. In modo simile, la j-esima autocovarianza è la covarianza tra Y_t e Y_{t-j}. [55]

L'autocovarianza e l'autocorrelazione j-esime possono essere stimate con l'autocovarianza e l'autocorrelazione campionarie j-esime: $\widehat{cov(Y_{t-j})}$ e $\hat{\rho}_j$.

[54] Stock, J. H., & Watson, M. W. (2020). *Introduzione all'econometria*. Quinta edizione. Pearson Italia Spa.
[55] Stock, J. H., & Watson, M. W. (2020). *Introduzione all'econometria*. Quinta edizione. Pearson Italia Spa.

$$\widehat{cov(Y_t, Y_{t-j})} = \frac{1}{T}\sum_{t=j+1}^{T}(Y_t - \bar{Y}_{j+1,T})(Y_{t-j} - \bar{Y}_{j-1,T})$$

$$\hat{\rho}_j = \frac{\widehat{cov(Y_t, Y_{t-j})}}{\widehat{var(Y_t)}}$$

Dove:

- $\bar{Y}_{j+1,T}$ indica la media campionaria di Y_t, calcolata sulle osservazioni $t = j+1, \dots, T$
- $\widehat{var(Y_t)}$ indica la varianza campionaria di Y

Una caratteristica importante delle serie storiche è la stazionarietà.
Una serie storica viene infatti definita stazionaria se, nell'ambito delle regressioni temporali, i modelli costruiti utilizzando i valori passati del fenomeno sono utili per prevederne quelli futuri. Più specificatamente, una serie è stazionaria se la sua distribuzione di probabilità non cambia nel corso del tempo. In caso contrario la serie viene detta "non stazionaria" oppure "con radice unitaria". I due principali tipi di non stazionarietà sono i *trend* ed i *break* strutturali.

1. *Trend*: distinguiamo tra trend deterministico (funzione deterministica del tempo come per esempio $y_t = t^2$) e trend stocastico (*trend* casuale che varia nel tempo). L'esempio principale di *trend* stocastico è la *random walk*: un caso particolare di $AR(1)$, che approfondiremo in seguito, con $\beta_0 = 0$ e $\beta_1 = 1$, dove la migliore previsione di Y_{t+1} è il suo valore al tempo t. La *random walk* con *drift*, invece, segue una passeggiata aleatoria attorno a un *trend* lineare e la migliore previsione di Y_{t+1} è uguale a $\beta_0 + Y_t$.
2. *Break* strutturali: si hanno quando i coefficienti del modello non sono costanti sull'intero campione. Ci si riferisce quindi ad una rottura o discontinuità nel comportamento di una serie storica che può essere causata da diversi fattori, come eventi imprevisti, cambiamenti nella struttura sottostante dei dati o interventi umani.

Lo studio delle serie storiche è il più delle volte finalizzato alla costruzione di un modello di previsione. La stima di una previsione è indicata con $\hat{Y}_{T+1|T}$, dove:

- $T+1|T$ indica che la previsione riguarda il valore di Y al tempo $T+1$ calcolata usando tutti i dati fino al tempo T.
- ^ indica che la previsione si basa su un modello stimato.

Pertanto, poiché nessun modello è in grado di prevedere al 100% il futuro, è inevitabile commettere qualche errore. Questi errori sono chiamati "errore di previsione" e sono dati dalla differenza tra il valore effettivamente osservato al tempo $T + 1$ e quello che si era precedentemente stimato.
Al fine di ottenere il modello più preciso possibile, lo scopo degli studiosi è quello di minimizzare il più possibile gli errori. La misura quantitativa più comunemente usata per analizzare gli errori di previsione è l'errore di previsione quadratico medio, o anche noto come MSFE (*Mean Squared Forecast Error*):

$$MSFE = E[(Y_{T+1} - \hat{Y}_{T+1|T})^2]$$

L'$MSFE$ è un indicatore molto utile in quanto, elevando al quadrato i singoli errori di previsione, mette in risalto la presenza di un possibile errore molto grande che può compromettere l'intera previsione e minimizza invece tutti quei possibili piccoli errori che risultano insignificanti.
Per ottenere invece un indice che utilizzi la stessa unità di Y, si usa il $RMSFE$(*Root Mean Squared Forecast Error*), che si calcola come radice quadrata dell'$MSFE$.

4.1.1 MODELLO AUTOREGRESSIVO DI ORDINE p

Il modello autoregressivo di ordine p, o più semplicemente noto come $AR(p)$, è un tipo di modello statistico utilizzato per analizzare le serie temporali e si usa per prevedere i valori futuri utilizzando i soli valori passati. Questo tipo di modello è particolarmente utile quando i dati mostrano una certa forma di dipendenza temporale.
Nel modello $AR(p)$, p rappresenta l'ordine del modello, ovvero il numero di valori passati che si intendono considerare per effettuare la previsione del valore attuale. Ad esempio, se $p = 1$, il modello considera solo l'ultimo valore osservato per fare la previsione, mentre se $p = 2$, considera gli ultimi due valori. Questo modello rappresenta quindi Y_t come una funzione lineare dei suoi p ritardi:

$$Y_t = \beta_0 + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \cdots + \beta_p Y_{t-p} + u_t$$

Dove:

- β_0 è una costante e $\beta_1, \beta_2, \dots, \beta_p$ sono i coefficienti autoregressivi, che misurano l'effetto dei valori precedenti sulla previsione. Tutti questi coefficienti possono essere stimati mediante i minimi quadrati ordinari (OLS).
- $Y_{t-1}, Y_{t-2}, \dots, Y_{t-p}$ sono i valori ritardati di Y.
- u_t è il residuo, che rappresenta la parte non spiegata dal modello.

Si assume, inoltre, che u_t abbia media nulla condizionatamente ai valori passati di Y_t:

$$E(u_t|Y_{t-1}, Y_{t-2}, \dots) = 0$$

Per verificare l'ipotesi che $Y_{t-1}, Y_{t-2}, \dots, Y_{t-p}$ non sono utili per prevedere Y_t, serve un test congiunto sui coefficienti $\beta_1, \beta_2, \dots, \beta_p$: il test *F.*

Per calcolare la bontà del modello autoregressivo proposto si utilizza l'R^2 *adjusted* che, a differenza dell'R^2 classico, tiene conto del numero di regressori utilizzati:

$$R^2_{adj} = 1 - \frac{n-1}{n-k-1}(1 - R^2)$$

Dove n è il numero totale di osservazioni e k è il numero di regressori utilizzati nel modello, inclusa la costante.
Un problema che sorge durante la costruzione di un modello autoregressivo è la scelta del numero di regressori p. Per determinarne il valore ottimale, si valuta la frequenza e la numerosità dei dati e ci si avvale dei cosiddetti "criteri di informazione": AIC, BIC e HQIC.
Questi tre criteri d'informazione vengono utilizzati per selezionare il miglior modello tra un insieme di modelli candidati e differiscono soltanto nella loro formulazione matematica e nell'approccio alla penalizzazione della complessità del modello.

1. BIC - *Bayes Information Criterion*
   Questo criterio di informazione tende a preferire modelli più semplici e quindi con pochi regressori.
   $$BIC(p) = \ln\left(\frac{SSR(p)}{T}\right) + (p+1)\frac{\ln(T)}{T}$$

2. AIC – *Akaike Information Criterion*
   Questo criterio di informazione tende invece a preferire modelli più complessi e quindi con molti regressori.
   $$AIC(p) = \ln\left(\frac{SSR(p)}{T}\right) + (p+1)\frac{2}{T}$$

3. HQIC - *Hannan-Quinn Information Criterion*

   Quest'ultimo criterio di informazione si configura invece come una "via di mezzo" tra i due visti in precedenza.

$$HQIC(p) = \ln\left(\frac{SSR(p)}{T}\right) + (p+1)\frac{2\ln(\ln(T))}{T}$$

Qualsiasi criterio si utilizza, l'obiettivo è sempre quello di minimizzarne il valore andando a scegliere quindi sempre il modello che presenta il criterio di informazione più basso.

4.1.2 MODELLO AUTOREGRESSIVO MISTO

Il modello autoregressivo misto, anche noto come ADL (*Autoregressive Distributed Lag*), si differenzia dal modello analizzato in precedenza per il fatto che, per stimare i valori futuri, oltre ad utilizzare i ritardi della variabile dipendente, utilizza anche i ritardi di un predittore aggiuntivo. Pertanto, il modello autoregressivo misto con p ritardi della variabile Y_t e q ritardi del predittore X_t è detto modello $ADL(p,q)$ ed è rappresentato dalla seguente equazione[56]:

$$\begin{aligned} Y_t = {} & \beta_0 + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \cdots + \beta_p Y_{t-p} \\ & + \delta_1 X_{t-1} + \delta_2 X_{t-2} + \cdots + \delta_q X_{t-q} + u_t \end{aligned}$$

È possibile, inoltre, aggiungere anche più di un predittore addizionale.

Il modello generale di regressione temporale con k predittori addizionali e q_1 ritardi del primo predittore, q_2 ritardi del secondo predittore, e così via, è rappresentato invece dalla seguente equazione[57]:

$$\begin{aligned} Y_t = {} & \beta_0 + \beta_1 Y_{t-1} + \beta_2 Y_{t-2} + \cdots + \beta_p Y_{t-p} \\ & + \delta_{11} X_{1t-1} + \delta_{12} X_{1t-2} + \cdots + \delta_{1q_1} X_{1t-q_1} \\ & + \cdots + \delta_{k1} X_{kt-1} + \delta_{k2} X_{kt-2} + \cdots + \delta_{kq_k} X_{kt-q_k} + u_t \end{aligned}$$

Dove:

- $E(u_t | Y_{t-1}, Y_{t-2}, \ldots, X_{1t-1}, X_{1t-2}, \ldots, X_{kt-1}, X_{kt-2}, \ldots) = 0$
- Le variabili aleatorie $(Y_t, X_{1t}, \ldots, X_{kt})$ hanno una distribuzione stazionaria.
- $(Y_t, X_{1t}, \ldots, X_{kt})$ e $(Y_{t-j}, X_{1t-j}, \ldots, X_{kt-j})$ diventano indipendenti al crescere di j.
- Gli *outlier* sono improbabili: $X_{1t}, \ldots, X_{kt}$ e Y_t hanno momenti quarti finiti diversi da zero.
- Non c'è perfetta collinearità.

[56] Stock, J. H., & Watson, M. W. (2020). *Introduzione all'econometria*. Quinta edizione. Pearson Italia Spa.
[57] Stock, J. H., & Watson, M. W. (2020). *Introduzione all'econometria*. Quinta edizione. Pearson Italia Spa.

Così come nei modelli $AR(p)$, anche nei modelli autoregressivi misti si utilizzano i tre criteri di informazione visti in precedenza che, dato K pari alla somma dei coefficienti (inclusa l'intercetta), diventano:

- $BIC(K) = \ln\left(\frac{SSR(K)}{T}\right) + \frac{\mathrm{K}\ln(T)}{T}$
- $AIC(K) = \ln\left(\frac{SSR(K)}{T}\right) + \frac{2K}{T}$
- $HQIC(K) = \ln\left(\frac{SSR(K)}{T}\right) + \frac{2\mathrm{K}\ln(\ln(T))}{T}$

4.2 APPLICAZIONE CON R

Nel seguente paragrafo verranno illustrati tutti i comandi utilizzati in R per effettuare le analisi statistiche sui set di dati estrapolati in precedenza da *Refinitiv* e *Google Trends,* al fine di verificare se attraverso la *sentiment analysis* è possibile predire i rendimenti dei titoli. In particolare, vedremo come evolverà l'R^2 con i diversi modelli di autoregressione.

Il primo step che ho effettuato è stato quello di scaricare tutte le librerie utili all'analisi dei dati.

```
library(xts)
library(highfrequency)
library(tidyquant)
library(lmtest)
library(sandwich)
library(readxl)
```

Fig.30 – Librerie

Tramite la funzione "*read_excel()*" ho poi importato il file Excel in R in modo da poter esportare e rinominare le singole colonne in formato "xts".

```
#Leggo file excel
amazon = read_excel("dati_amzn.xlsx")

#Esporto le singole colonne in formato xts
rend_amzn = xts(as.numeric(amazon$Rend_perc), as.Date(amazon$Data, format = "%Y-%m-%d"))
sentiment_amzn = xts(as.numeric(amazon$Sentiment), as.Date(amazon$Data, format = "%Y-%m-%d"))
googletrends_amzn = xts(as.numeric(amazon$`Google Trend`), as.Date(amazon$Data, format = "%Y-%m-%d"))

#Rinomino le colonne
colnames(rend_amzn) = "rend_amzn"
colnames(sentiment_amzn) = "sentiment_amzn"
colnames(googletrends_amzn) = "googletrends_amzn"
```

Fig.31 – Lettura del file Excel

Avendo estrapolato le serie storiche dei rendimenti, del *sentiment* e dei volumi di ricerca su Google, è possibile iniziare a creare i modelli autoregressivi partendo da $AR(1)$, $AR(2)$ e $AR(3)$.

```
# MODELLI AR -------------------------------
#AR(1)
rendimenti_amzn = rend_amzn[2:307]
rend_amzn_1 = rend_amzn[1:306]

names(rendimenti_amzn) = "rendimenti_amzn"
names(rend_amzn_1) = "rend_amzn_1"

ar_1_amzn =
  lm(rendimenti_amzn ~ rend_amzn_1,
     data.frame(coredata(rendimenti_amzn['2022/2023']),
                coredata(rend_amzn_1['2022/2023'])))
summary(ar_1_amzn)

rmsfe_ar_1_amzn = sqrt(mean((rendimenti_amzn - fitted(ar_1_amzn))^2))

#AR(2)
rendimenti_amzn = rend_amzn[3:307]
rend_amzn_1 = rend_amzn[1:305]
rend_amzn_2 = rend_amzn[2:306]

names(rendimenti_amzn) = "rendimenti_amzn"
names(rend_amzn_1) = "rend_amzn_1"
names(rend_amzn_2) = "rend_amzn_2"

ar_2_amzn =
  lm(rendimenti_amzn ~ rend_amzn_1 + rend_amzn_2,
     data.frame(coredata(rendimenti_amzn['2022/2023']),
                coredata(rend_amzn_1['2022/2023']),
                coredata(rend_amzn_2['2022/2023'])))
summary(ar_2_amzn)

rmsfe_ar_2_amzn = sqrt(mean((rendimenti_amzn - fitted(ar_2_amzn))^2))

#AR(3)
rendimenti_amzn = rend_amzn[4:307]
rend_amzn_1 = rend_amzn[1:304]
rend_amzn_2 = rend_amzn[2:305]
rend_amzn_3 = rend_amzn[3:306]

names(rendimenti_amzn) = "rendimenti_amzn"
names(rend_amzn_1) = "rend_amzn_1"
names(rend_amzn_2) = "rend_amzn_2"
names(rend_amzn_3) = "rend_amzn_3"

ar_3_amzn =
  lm(rendimenti_amzn ~ rend_amzn_1 + rend_amzn_2 + rend_amzn_3,
     data.frame(coredata(rendimenti_amzn['2022/2023']),
                coredata(rend_amzn_1['2022/2023']),
                coredata(rend_amzn_2['2022/2023']),
                coredata(rend_amzn_3['2022/2023'])))
summary(ar_3_amzn)

rmsfe_ar_3_amzn = sqrt(mean((rendimenti_amzn - fitted(ar_3_amzn))^2))
```

Fig.32 – Modelli AR

Come vedremo in seguito nella tabella riassuntiva degli R^2 dei modelli autoregressivi sui rendimenti dei titoli di Amazon, questi sono estremamente bassi per i modelli $AR(p)$. Questo sta a significare che, come facilmente prevedibile, è impossibile predire l'andamento dei rendimenti soltanto sulla base dei valori passati perché questi ultimi seguono un andamento *random walk,* quindi casuale.

Ho poi creato i modelli $ADL(2,1)$, $ADL(2,2)$ e $ADL(1,2)$ con regressore aggiuntivo "Google Trends" scrivendo il seguente codice:

```
# MODELLI ADL CON GOOGLE-------------------------------
# ADL(2,1)
rendimenti_amzn = rend_amzn[3:307]
rend_amzn_1 = rend_amzn[1:305]
rend_amzn_2 = rend_amzn[2:306]
googletrends_amzn_1 = googletrends_amzn[1:305]

names(rendimenti_amzn) = "rendimenti_amzn"
names(rend_amzn_1) = "rend_amzn_1"
names(rend_amzn_2) = "rend_amzn_2"
names(googletrends_amzn_1) = "googletrens_amzn_1"

adl_21_amzn_google =
  lm(rendimenti_amzn ~ rend_amzn_1 + rend_amzn_2 + googletrends_amzn_1,
     data.frame(coredata(rendimenti_amzn['2022/2023']),
                coredata(rend_amzn_1['2022/2023']),
                coredata(rend_amzn_2['2022/2023']),
                coredata(googletrends_amzn_1['2022/2023'])))
summary(adl_21_amzn_google)

rmsfe_adl_21_amzn_google = sqrt(mean((rendimenti_amzn - fitted(adl_21_amzn_google))^2))

# ADL(2,2)
rendimenti_amzn = rend_amzn[3:307]
rend_amzn_1 = rend_amzn[1:305]
rend_amzn_2 = rend_amzn[2:306]
googletrends_amzn_1 = googletrends_amzn[1:305]
googletrends_amzn_2 = googletrends_amzn[2:306]

names(rendimenti_amzn) = "rendimenti_amzn"
names(rend_amzn_1) = "rend_amzn_1"
names(rend_amzn_2) = "rend_amzn_2"
names(googletrends_amzn_1) = "googletrens_amzn_1"
names(googletrends_amzn_2) = "googletrens_amzn_2"

adl_22_amzn_google =
  lm(rendimenti_amzn ~ rend_amzn_1 + rend_amzn_2 + googletrends_amzn_1 + googletrends_amzn_2,
     data.frame(coredata(rendimenti_amzn['2022/2023']),
                coredata(rend_amzn_1['2022/2023']),
                coredata(rend_amzn_2['2022/2023']),
                coredata(googletrends_amzn_1['2022/2023']),
                coredata(googletrends_amzn_2['2022/2023'])))
summary(adl_22_amzn_google)

rmsfe_adl_22_amzn_google = sqrt(mean((rendimenti_amzn - fitted(adl_22_amzn_google))^2))

# ADL(1,2)
rendimenti_amzn = rend_amzn[3:307]
rend_amzn_1 = rend_amzn[1:305]
googletrends_amzn_1 = googletrends_amzn[1:305]
googletrends_amzn_2 = googletrends_amzn[2:306]

names(rendimenti_amzn) = "rendimenti_amzn"
names(rend_amzn_1) = "rend_amzn_1"
names(googletrends_amzn_1) = "googletrens_amzn_1"
names(googletrends_amzn_2) = "googletrens_amzn_2"

adl_12_amzn_google =
  lm(rendimenti_amzn ~ rend_amzn_1 + googletrends_amzn_1 + googletrends_amzn_2,
     data.frame(coredata(rendimenti_amzn['2022/2023']),
                coredata(rend_amzn_1['2022/2023']),
                coredata(googletrends_amzn_1['2022/2023']),
                coredata(googletrends_amzn_2['2022/2023'])))
summary(adl_12_amzn_google)

rmsfe_adl_12_amzn_google = sqrt(mean((rendimenti_amzn - fitted(adl_12_amzn_google))^2))

#MODELLI AUTOREGRESSIVI MISTI CON GOOGLE E SENTIMENT-------------------------------
#111
rendimenti_amzn = rend_amzn[2:307]
rend_amzn_1 = rend_amzn[1:306]
googletrends_amzn_1 = googletrends_amzn[1:306]
sentiment_amzn_1 = sentiment_amzn[1:306]

names(rendimenti_amzn) = "rendimenti_amzn"
names(rend_amzn_1) = "rend_amzn_1"
names(googletrends_amzn_1) = "googletrens_amzn_1"
names(sentiment_amzn_1) = "sentiment_amzn_1"

adl_111_amzn = lm(rendimenti_amzn ~ rend_amzn_1 + googletrends_amzn_1 + sentiment_amzn_1,
                  data.frame(coredata(rendimenti_amzn['2022/2023']),
                             coredata(rend_amzn_1['2022/2023']),
                             coredata(googletrends_amzn_1['2022/2023']),
                             coredata(sentiment_amzn_1['2022/2023'])))
summary(adl_111_amzn)

rmsfe_adl_111_amzn = sqrt(mean((rendimenti_amzn - fitted(adl_111_amzn))^2))
```

Fig.33 – Modelli ADL

Infine, per ottenere un modello autoregressivo ancora più preciso, ho effettuato un'autoregressione mista con due parametri aggiuntivi, "Google Trends" e *sentiment*, prima con uno, poi con due ed infine con tre ritardi:

```
#MODELLI AUTOREGRESSIVI MISTI CON GOOGLE E SENTIMENT------------------------------
#111
rendimenti_amzn = rend_amzn[2:307]
rend_amzn_1 = rend_amzn[1:306]
googletrends_amzn_1 = googletrends_amzn[1:306]
sentiment_amzn_1 = sentiment_amzn[1:306]

names(rendimenti_amzn) = "rendimenti_amzn"
names(rend_amzn_1) = "rend_amzn_1"
names(googletrends_amzn_1) = "googletrens_amzn_1"
names(sentiment_amzn_1) = "sentiment_amzn_1"

adl_111_amzn = lm(rendimenti_amzn ~ rend_amzn_1 + googletrends_amzn_1 + sentiment_amzn_1,
                  data.frame(coredata(rendimenti_amzn['2022/2023']),
                             coredata(rend_amzn_1['2022/2023']),
                             coredata(googletrends_amzn_1['2022/2023']),
                             coredata(sentiment_amzn_1['2022/2023'])))
summary(adl_111_amzn)

rmsfe_adl_111_amzn = sqrt(mean((rendimenti_amzn - fitted(adl_111_amzn))^2))

#222
rendimenti_amzn = rend_amzn[3:307]
rend_amzn_1 = rend_amzn[1:305]
rend_amzn_2 = rend_amzn[2:306]
googletrends_amzn_1 = googletrends_amzn[1:305]
googletrends_amzn_2 = googletrends_amzn[2:306]
sentiment_amzn_1 = sentiment_amzn[1:305]
sentiment_amzn_2 = sentiment_amzn[2:306]

names(rendimenti_amzn) = "rendimenti_amzn"
names(rend_amzn_1) = "rend_amzn_1"
names(rend_amzn_2) = "rend_amzn_2"
names(googletrends_amzn_1) = "googletrens_amzn_1"
names(googletrends_amzn_2) = "googletrens_amzn_2"
names(sentiment_amzn_1) = "sentiment_amzn_1"
names(sentiment_amzn_2) = "sentiment_amzn_2"

adl_222_amzn = lm(rendimenti_amzn ~ rend_amzn_1 + rend_amzn_2 +
                    googletrends_amzn_1 + googletrends_amzn_2 +
                    sentiment_amzn_1 + sentiment_amzn_2,
                  data.frame(coredata(rendimenti_amzn['2022/2023']),
                             coredata(rend_amzn_1['2022/2023']),
                             coredata(rend_amzn_2['2022/2023']),
                             coredata(googletrends_amzn_1['2022/2023']),
                             coredata(googletrends_amzn_2['2022/2023']),
                             coredata(sentiment_amzn_1['2022/2023']),
                             coredata(sentiment_amzn_2['2022/2023'])))
summary(adl_222_amzn)

rmsfe_adl_222_amzn = sqrt(mean((rendimenti_amzn - fitted(adl_222_amzn))^2))

#333
rendimenti_amzn = rend_amzn[4:307]
rend_amzn_1 = rend_amzn[1:304]
rend_amzn_2 = rend_amzn[2:305]
rend_amzn_3 = rend_amzn[3:306]
googletrends_amzn_1 = googletrends_amzn[1:304]
googletrends_amzn_2 = googletrends_amzn[2:305]
googletrends_amzn_3 = googletrends_amzn[3:306]
sentiment_amzn_1 = sentiment_amzn[1:304]
sentiment_amzn_2 = sentiment_amzn[2:305]
sentiment_amzn_3 = sentiment_amzn[3:306]

names(rendimenti_amzn) = "rendimenti_amzn"
names(rend_amzn_1) = "rend_amzn_1"
names(rend_amzn_2) = "rend_amzn_2"
names(rend_amzn_3) = "rend_amzn_3"
names(googletrends_amzn_1) = "googletrens_amzn_1"
names(googletrends_amzn_2) = "googletrens_amzn_2"
names(googletrends_amzn_3) = "googletrens_amzn_3"
names(sentiment_amzn_1) = "sentiment_amzn_1"
names(sentiment_amzn_2) = "sentiment_amzn_2"
names(sentiment_amzn_3) = "sentiment_amzn_3"

adl_333_amzn = lm(rendimenti_amzn ~ rend_amzn_1 + rend_amzn_2 + rend_amzn_3 +
                    googletrends_amzn_1 + googletrends_amzn_2 + googletrends_amzn_3 +
                    sentiment_amzn_1 + sentiment_amzn_2 + sentiment_amzn_3,
                  data.frame(coredata(rendimenti_amzn['2022/2023']),
                             coredata(rend_amzn_1['2022/2023']),
                             coredata(rend_amzn_2['2022/2023']),
                             coredata(rend_amzn_3['2022/2023']),
                             coredata(googletrends_amzn_1['2022/2023']),
                             coredata(googletrends_amzn_2['2022/2023']),
                             coredata(googletrends_amzn_3['2022/2023']),
                             coredata(sentiment_amzn_1['2022/2023']),
                             coredata(sentiment_amzn_2['2022/2023']),
                             coredata(sentiment_amzn_3['2022/2023'])))
summary(adl_333_amzn)

rmsfe_adl_333_amzn = sqrt(mean((rendimenti_amzn - fitted(adl_333_amzn))^2))
```

Fig.34 – Modelli autoregressivi misti con Google trends e *sentiment*

Per concludere, attraverso quest'ultima riga di comando, ho stampato la tabella riassuntiva dell'andamento dell'R^2:

```
#TABELLA RIASSUNTIVA--------------------------------
riassunto_rsquared_amzn = data.frame(Modello = c('AR(1)',
                                                 'AR(2)',
                                                 'AR(3)',
                                                 'ADL(2,1) Google',
                                                 'ADL(2,2) Google',
                                                 'ADL(1,2) Google',
                                                 '1 ritardo di Google e Sentiment',
                                                 '2 ritardi di Google e Sentiment',
                                                 '3 ritardi di Google e Sentiment'),
                                     R_squared = c(summary(ar_1_amzn)$r.squared,
                                                   summary(ar_2_amzn)$r.squared,
                                                   summary(ar_3_amzn)$r.squared,
                                                   summary(adl_21_amzn_google)$r.squared,
                                                   summary(adl_22_amzn_google)$r.squared,
                                                   summary(adl_12_amzn_google)$r.squared,
                                                   summary(adl_111_amzn)$r.squared,
                                                   summary(adl_222_amzn)$r.squared,
                                                   summary(adl_333_amzn)$r.squared),
                                     AIC = c(AIC(ar_1_amzn), AIC(ar_2_amzn), AIC(ar_3_amzn),
                                             AIC(adl_21_amzn_google), AIC(adl_22_amzn_google),
                                             AIC(adl_12_amzn_google), AIC(adl_111_amzn),
                                             AIC(adl_222_amzn), AIC(adl_333_amzn)),
                                     RMSFE = c(rmsfe_ar_1_amzn, rmsfe_ar_2_amzn, rmsfe_ar_3_amzn,
                                               rmsfe_adl_21_amzn_google, rmsfe_adl_22_amzn_google,
                                               rmsfe_adl_12_amzn_google, rmsfe_adl_111_amzn,
                                               rmsfe_adl_222_amzn, rmsfe_adl_333_amzn))
```

Fig.35 – Tabella riassuntiva

Ottenendo, per quanto riguarda Amazon, il seguente *output:*

Modello	R_squared	AIC	RMSFE
AR(1)	0.0003089001	1553.443	3.033032
AR(2)	0.0040726663	1550.223	3.032222
AR(3)	0.0046042102	1541.431	3.003755
ADL(2,1) Google	0.0059761792	1551.639	3.029323
ADL(2,2) Google	0.0061289808	1553.592	3.029090
ADL(1,2) Google	0.0057988571	1551.694	3.029593
1 ritardo di Google e Sentiment	0.0034054476	1556.494	3.028331
2 ritardi di Google e Sentiment	0.0121339679	1555.744	3.019925
3 ritardi di Google e Sentiment	0.0179333093	1549.333	2.983576

Fig.36 – Modelli autoregressivi (Amazon)

È molto chiaro da questa tabella come, prima con l'inserimento dei volumi di ricerca su Google e poi con l'inserimento dei valori del *sentiment*, l'R^2 aumenti significativamente passando orientativamente dallo 0,031% all' 1,79%.

Osservando le tabelle riassuntive di Tesla (Fig.37) e di Apple (Fig.38) è possibile notare che è stato raggiunto circa lo stesso risultato arrivando addirittura ad un $R^2 = 2{,}58\%$ in Apple.

Modello	R_squared	AIC	RMSFE
AR(1)	0.002624586	1739.722	4.112124
AR(2)	0.005422363	1735.790	4.110333
AR(3)	0.006167856	1732.765	4.114659
ADL(2,1) Google	0.009939488	1736.402	4.100989
ADL(2,2) Google	0.012248734	1737.689	4.096203
ADL(1,2) Google	0.010584654	1736.203	4.099652
1 ritardo di Google e Sentiment	0.008278326	1741.982	4.100453
2 ritardi di Google e Sentiment	0.020712720	1739.065	4.078615
3 ritardi di Google e Sentiment	0.024624351	1739.066	4.076273

Fig.37 – Modelli autoregressivi (Tesla)

Modello	R_squared	AIC	RMSFE
AR(1)	0.001510465	1327.361	2.096246
AR(2)	0.010161507	1323.276	2.090182
AR(3)	0.010478352	1321.229	2.091096
ADL(2,1) Google	0.010161756	1325.276	2.090181
ADL(2,2) Google	0.011604887	1326.831	2.088657
ADL(1,2) Google	0.009621433	1325.442	2.090752
1 ritardo di Google e Sentiment	0.005169250	1330.238	2.092402
2 ritardi di Google e Sentiment	0.023219932	1327.225	2.076348
3 ritardi di Google e Sentiment	0.025779194	1328.492	2.074866

Fig.38 – Modelli autoregressivi (Apple)

4.3 COMMENTO DEI RISULTATI OTTENUTI

Nonostante questi valori dell'R^2 possano sembrare molto bassi e poco significativi, analizziamo i risultati ottenuti per comprenderli al meglio:

- $AR(p)$

  Per tutti i titoli presi in considerazione, come spiegato in precedenza e come facilmente intuibile, questo tipo di modello non è assolutamente adatto per stimarne i rendimenti futuri;

infatti, i valori dell' R^2 aumentano all'aumentare dei ritardi ma restano comunque molto piccoli.

- $ADL(p,q)$ con Google Trends
  Passando al modello autoregressivo misto, utilizzando come predittore aggiuntivo Google Trends, vediamo che l'R^2 aumenta all'aumentare di q nei modelli $ADL(2,1)$ e $ADL(2,2)$ ed invece diminuisce se si utilizza un solo ritardo dei rendimenti, quindi nel modello $ADL(1,2)$. Generalizzando possiamo affermare che questo tipo di regressore aggiuntivo è utile per migliorare la stima dei rendimenti futuri ma che comunque è necessario utilizzare più di un ritardo dei rendimenti in quanto il ritardo primo è molto poco significativo

- Modello autoregressivo misto con Google Trends e *sentiment*
  In quest'ultimo step è stato creato un modello autoregressivo misto migliorativo di quello precedente perché impreziosito dalla presenza dei dati del *sentiment*. Come detto nel punto precedente, l'utilizzo di un solo ritardo porta ad ottenere un R^2 molto basso e questa tendenza si conferma anche in questo modello. Quando però si costruisce questo modello utilizzando prima due e poi tre ritardi, è possibile notare come finalmente il valore dell'R^2 cominci a crescere significativamente.

I valori finali dei migliori R^2, ottenuti tutti dall'ultimo modello costruito, sono circa 1,79% per Amazon, 2,46% per Tesla e 2,58% per Apple.
In generale questi risultati sono comunque molto bassi però, prima di giungere a conclusioni affrettate, è opportuno calarsi nel contesto che stiamo analizzando.
In primis è chiaro che, se compariamo l'R^2 ottenuto dal primo modello $AR(1)$ con quello ottenuto dall'ultimo modello, abbiamo un miglioramento, calcolato come rapporto tra valore finale e valore iniziale, del 5.805,54% per Amazon, 938,22% per Tesla e del 1.706,71% per Apple.
C'è poi da considerare il fatto che i rendimenti giornalieri dei titoli azioni seguono una *random walk* e proprio per questo sono quasi impossibili da prevedere; se infatti fosse possibile prevederli si potrebbe "battere il mercato" ed ottenere sistematicamente degli extra-profitti.
Quindi, proprio perché i rendimenti seguono un andamento casuale, ogni giorno si ha il 50% di probabilità che il prezzo salga ed il 50% di probabilità che il prezzo diminuisca.

È bene ricordare che non esiste un metodo universale per individuare il miglior modello, ma quest'ultimo si può stimare a seconda del parametro che vogliamo prendere in considerazione; se

consideriamo l'R^2 bisogna scegliere il modello con il valore più alto, se invece consideriamo l'AIC o il RMSFE bisogna scegliere il modello con il valore più basso. Dando uno sguardo ai risultati ottenuti, i modelli consigliati sono i seguenti:

- Amazon: modello autoregressivo misto con tre ritardi, in quanto presenta l'R^2 più alto e l'AIC ed il RMSFE più bassi.
- Tesla: secondo l'R^2 ed il RMSFE il modello più accurato è quello misto con tre ritardi, mentre secondo l'AIC è l'$AR(3)$.
- Apple: in modo identico al caso precedente, secondo l'R^2 ed il RMSFE il modello più accurato è quello misto con tre ritardi, mentre secondo l'AIC è l'$AR(3)$.

CONCLUSIONI

Il presente lavoro è nato con l'obiettivo di riscontrare l'esistenza di una correlazione tra il *sentiment* ed il rendimento dei titoli azionari. Il tutto è partito dalla mia curiosità di verificare se, attraverso i sempre più potenti strumenti statistici ed informatici, fosse possibile in qualche modo "battere" il mercato riuscendo a prevedere anche solo una piccola parte delle variazioni future dei titoli.

A tal proposito abbiamo analizzato, e poi criticato, la *EMH Theory* secondo la quale non è possibile in alcun modo prevedere il mercato e tutte le risorse destinate alle analisi tecniche sono sprecate. Abbiamo poi approfondito il complesso mondo del *text mining* che si pone alla base del funzionamento della sentiment analysis, per poi estrarre tutti i dati utili alla nostra ricerca da Refinitiv Workspace e Google Trends. L'ultimo e più importante step è consistito nella creazione di modelli autoregressivi e nell'evidenziare come il *sentiment* del mercato e l'interesse verso i titoli abbiano migliorato l'accuratezza della capacità previsiva dei modelli.

Nell'ambito di questa ricerca, siamo riusciti, con l'introduzione del *sentiment* e dei dati di Google Trends, che rappresentano l'interesse delle persone, a spiegare circa il 2% delle variazioni dei titoli, rompendo anche se di poco l'equilibrio del 50 e 50 che avevamo in precedenza.
In conclusione, è possibile affermare che il *sentiment* è un predittore che spiega soltanto una piccola parte della variazione dei rendimenti ma che, se inserito all'interno di un'autoregressione temporale con altri predittori significativi, può risultare molto utile. Pertanto, il presente lavoro può essere un'ottima base di partenza per costruire modelli autoregressivi ancora più precisi ed affidabili.

BIBLIOGRAFIA

Ariel, R. A. (1987). A monthly effect in stock returns. *Journal of financial economics*, *18*(1), 161-174.

Barberis, N. & Thaler, R. (2003). A survey of behavioral finance. *Handbook of the Economics of Finance*, *1*, 1053-1128.

Barone, E. (1990). The Italian stock market: efficiency and calendar anomalies. *Journal of Banking & Finance*, *14*(2-3).

Barone, R. (2003). From efficient markets to behavioral finance.

Bayes, T. (1763). LII. An essay towards solving a problem in the doctrine of chances. S. *Philosophical transactions of the Royal Society of London*, (53), 370-418.

Black F. (1986). *Noise*. The Journal of Finance, Vol. 41, No. 3.

Bollen, J., Mao, H., & Zeng, X. (2011). Twitter mood predicts the stock market. *Journal of computational science*, 2(1), 1-8.

Borges, J. L. (1937-1952). The analytical language of John Wilkins. *Other inquisitions*.

Bouman, S., & Jacobsen, B. (2002). The Halloween indicator, "sell in May and go away": Another puzzle. *American Economic Review*, *92*(5), 1618-1635.

Cellino M. (2020). Il Sole 24 Ore, Cosa sono le vendite allo scoperto e perché vietarle (non sempre) funziona

Censis. 17º Rapporto sulla comunicazione, "I media dopo la pandemia". Roma, 6 ottobre 2021

Codice Civile.

Dizionari Simone.

Dizionario Treccani

Fama, E. F. (1965). The behavior of stock-market prices. *The journal of Business*.

Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The journal of Finance*.

French, K. R. (1980). Stock returns and the weekend effect. *Journal of financial economics*, *8*(1).

Gruhl, D., Guha, R., Kumar, R., Novak, J., & Tomkins, A. (2005) *The predictive power of online chatter*. (ACM, New York, NY, USA), pp. 78–87.

Jurafsky D. & Martin J.H. (2023). Speech and Language Processing. Third edition.

Kahneman D., Riepe. (1998). *The Psychology of non-Professional Investor*. Journal of Portfolio Management, Vol. 24, No.4

Keown, A. J., & Pinkerton, J. M. (1981). Merger announcements and insider trading activity: An empirical investigation. *The journal of finance*.

Mackay C. La pazzia delle folle Il Sole 24ore 2000.

Mishne, G & Glance, N. (2006) *Predicting Movie Sales from Blogger Sentiment*.

Mosteller, F., & Wallace, D. L. (1963). Inference in an authorship problem: A comparative study of discrimination methods applied to the authorship of the disputed Federalist Papers. *Journal of the American Statistical Association*, *58*(302), 275-309.

Pennebaker, J. W., Francis, M. E., & Booth, R. J. (2001). Linguistic inquiry and word count.

Porter, M. F. (1980). *An algorithm for suffix stripping. Program, 14(3), 130-137.*

Rozeff, M. S., & Kinney Jr, W. R. (1976). Capital market seasonality: The case of stock returns. *Journal of financial economics*, *3*(4).

Schwert, G. W. (2003). Anomalies and market efficiency. *Handbook of the Economics of Finance*, *1*.

Sharpe W. F. Alexander G. J. & Bailey J. V. (1990). *Investments William f. Sharpe Gordon j. Alexander fourth edition: instructor's manual*. Prentice Hall.

Shleifer, A. (2000). *Inefficient markets: An introduction to behavioral finance*. Oup Oxford.

Stock, J. H., & Watson, M. W. (2020). *Introduzione all'econometria*. Quinta edizione. Pearson

Stone, P. J., Dunphy, D. C., & Smith, M. S. (1966). The general inquirer: A computer approach to content analysis.

Weizenbaum, J. (1966). ELIZA - a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, *9*(1), 36-45.

Weizenbaum, J. (1976). Computer power and human reason: From judgment to calculation.

Wilson, T., Wiebe, J., & Hoffmann, P. (2005, October). Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of human language technology conference and conference on empirical methods in natural language processing* (pp. 347-354).

SITOGRAFIA

Bankpedia.org. ASSONEBB (Associazione Nazionale Enciclopedia della Banca e della Borsa)
Critiche all'efficienza di mercato.
https://blog.microfocus.com/how-much-data-is-created-on-the-internet-each-day/
https://www.ilpost.it/2020/01/15/diffusione-copie-quotidiani-2019/
https://www.refinitiv.com/en/about-us
https://www.statista.com/chart/26272/global-average-daily-time-spent-on-social-media-per-internet-user/
https://www.statista.com/statistics/1327116/number-of-global-tiktok-users/
https://www.statista.com/statistics/248769/age-distribution-of-worldwide-instagram-users/
https://www.statista.com/statistics/259477/hours-of-video-uploaded-to-youtube-every-minute/
https://www2.deloitte.com/content/dam/Deloitte/us/Documents/consumer-business/us-cb-navigating-the-new-digital-divide-051315.pdf